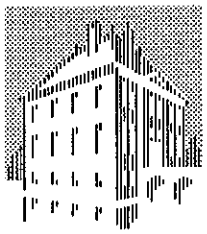
NOVEMBER 1992

FOM-INSTITUUT
VOOR
PLASMAFYSICA
RIJNHUIZEN



ASSOCIATIE
EURATOM-FOM

RELAX

A COMPUTER CODE FOR THE STUDY OF COLLISIONAL AND WAVE DRIVEN RELAXATION OF THE ELECTRON DISTRIBUTION FUNCTION IN TOROIDAL GEOMETRY

E. WESTERHOF, A.G. PEETERS, W.L. SCHIPPERS

RIJNHUIZEN REPORT 92-211

This work was performed as part of the research programme of the association agreement of Euratom and the 'Stichting voor Fundamenteel Onderzoek der Materie' (FOM) with financial support from the 'Nederlandse Organisatie voor Wetenschappelijk Onderzoek' (NWO) and Euratom.

POSTBUS 1207
3430 BE NIEUWEGEIN
NEDERLAND
EDISONBAAN 14
3439 MN NIEUWEGEIN
TEL. 03402 - 31224
TELEFAX 03402 -31204

ABSTRACT

The Fokker–Planck quasilinear code RELAX is described. The code solves the bounce-averaged Fokker–Planck equation for the evolution of the electron momentum distribution on a number of magnetic surfaces in a tokamak. The physics models incorporated in the code include bounce-averaged, approximate collision operators, electric field driven momentum space convection, and quasilinear diffusion due to electron cyclotron resonant heating. Interfaces are provided with the HELENA toroidal MHD equilibrium code (G.T.A. Huysmans et al., proceedings of the CP90 Europhysics Conference on *Computational Physics*, 10–13 September 1990, Amsterdam, The Netherlands, Editor A. Tenner, World Scientific, Singapore, (1991) p. 371) and with the TORAY ray-tracing code (A.H. Kritz et al., proceedings of the 3rd Intern. Symposium on Heating in Toroidal Plasmas, Grenoble (France), 22–26 March (1982) Vol. II, p. 707). A number of test cases are presented in which the code results are compared with known analytical results. The code will be used for the study of the generation and the behaviour of nonthermal electron populations in tokamak experiments. Another application of the code will be the study of non-inductive current drive by electron cyclotron waves.

The code RELAX is written as a driver for the FPPAC package developed at Livermore by M.G. McCoy et al. (*Comput. Phys. Commun.* **24** (1981) 37, and **51** (1988) 373).

ACKNOWLEDGEMENTS

During the past few years, we have benefitted greatly from numerous discussions with people, who are more experienced than us in the development and use of Fokker–Planck codes. In particular we want to thank Drs. G. Giruzzi, M.G. McCoy, R.W. Harvey, C.F.F. Carney, and G.R. Smith. We also want to thank Dr. T.J. Schep for his continuing interest in this work and stimulating discussions. Finally, we want to thank Dr. R.H. Cohen for making his code for the adjoint calculation of the current drive efficiency available to us.

TABLE OF CONTENTS

1. INTRODUCTION

The evolution, on a collisional timescale, of the particle distribution functions in a plasma is described by the Fokker–Planck equations. In this report, a description is given of the computer code RELAX, which has been written to solve the Fokker–Planck equation for the electrons in toroidal geometry. The core of the code is formed by FPPAC which was developed at Livermore by McCoy *et al.* [1,2] for the solution of the multispecies nonlinear Fokker–Planck equation. An excellent review of this code is given in the book by Killeen *et al.* [3]. In particular, the numerical core of FPPAC, responsible for the time advancement of the Fokker–Planck equation, has been left untouched. An important feature of FPPAC is the inclusion of the complete non-relativistic collision operator. However, the inhomogeneity of the magnetic field in toroidal geometry is not accounted for in FPPAC. In toroidal geometry the particles describe nearly periodic orbits along the field lines. In most cases of interest, the time between successive collisions is longer than the time required for the particles to complete one such orbit. As a consequence, the Fokker–Planck equation must be averaged over the particle orbits. This procedure is called bounce-averaging.

A new set of codes (CQL and CQL3D) has been developed by the same authors as FPPAC, in which the consequences of bounce-averaging are treated as complete as possible. An important drawback of such a complete treatment is the large amount of computing power that is required. For example, bounce-averaging of the complete collision operator can only be done numerically. In the development of the present code RELAX the emphasis has been to obtain a versatile code with minimum demands on required computing power. For this reason, simplified bounce-averaged collision, and wave diffusion operators are developed, retaining the essential physics with a minimum amount of computational effort.

The report consists of four parts. Firstly, the underlying theoretical framework is discussed in Section 2. This section presents the physics models used in the code, including the collison operator, the momentum space flux driven by a DC electric field, and the quasilinear diffusion due to electron cyclotron resonant

wave interactions. Next, the general structure of the code, the input and the output files are described in Section 3. In Section 4, a number of physical examples is treated, validating the implementation of the various physics models. Finally, some more detailed and technical descriptions of various aspects of the code are presented in the Appendices. These describe details of the numerical techniques, and of the interfacing with an MHD equilibrium code and an electron cyclotron ray-tracing code.

2. THEORETICAL FRAMEWORK

2.1 Charged particle orbits in toroidal geometry

In this section a brief summary of the motion of charged particles in a strong magnetic field with toroidal geometry is presented. It is assumed that the magnetic field lines form a set of closed, nested magnetic surfaces. In the case of a strong magnetic field the gyro-period and Larmor radius of the particle are much smaller than the timescale and lengthscale over which the magnetic field changes. The magnetic moment μ, which is defined as

$$\mu \equiv \frac{p_\perp^2}{2mB}, \tag{2.1.1}$$

where $p_\perp$ is the momentum perpendicular to the magnetic field B, is then an adiabatic constant of the motion, while the motion of the particle gyro-centre is approximately along the magnetic field line. Because of the conservation of the magnetic moment and the energy $\varepsilon \equiv p^2/2m$, the momentum parallel to the magnetic field can be expressed in terms of these conserved quantities μ and ε

$$p_\parallel = \mathrm{sgn}(p_\parallel)\,\sqrt{(\varepsilon - \mu B)2m}. \tag{2.1.2}$$

Two classes of particles exist: a class of circulating particles and a class of particles that is trapped between the maxima of B along the field line. Let B_0 and B_m be the minimum and maximum of B along the field line, respectively, and $p_{\parallel 0}$ and $p_{\perp 0}$ be the parallel and perpendicular momenta at the position of minimum B, then the particles for which

$$\frac{p_{\perp 0}^2}{p^2} \geq \frac{B_0}{B_m}, \tag{2.1.3}$$

are trapped, and describe periodic orbits between their turning points. Also the passing particles have nearly periodic orbits completing a full poloidal turn around

the magnetic surface. The time required to complete one such orbit, known as the bounce-period τ_B, and the associated bounce-frequency ω_B are given by

$$\tau_B = \frac{2\pi}{\omega_B} = \oint \frac{ds}{v\cos\theta} = \oint \frac{ds}{v_\parallel}, \tag{2.1.4}$$

where $\theta = \arccos p_\parallel/p$ is the pitch-angle and ds is the element of arclength along the magnetic field line associated with the gyro-centre motion. Note that ds is defined as positive for motion parallel to the magnetic field and negative for motion anti-parallel to B. One can define a bounce-phase ϕ_B by

$$d\phi_B = \omega_B \frac{ds}{v\cos\theta}. \tag{2.1.5}$$

There is also a second adiabatic invariant $J_\parallel$ that corresponds to the bounce-phase (*cf.* the magnetic moment μ and the gyro-phase),

$$J_\parallel = \oint p_\parallel ds. \tag{2.1.6}$$

The distribution function at a given magnetic surface is most conveniently written as a function of only these two adiabatic invariants. Equivalently, one can also use the momenta $p_{\parallel 0}$ and $p_{\perp 0}$, or the momentum p and the pitch-angle θ_0 at the position of minimum B along the field line instead of the invariants.

2.2 Bounce-averaged Fokker–Planck theory

Here, we closely follow the discussion presented in Chapter 3 of Ref. [3]. The general form of the Fokker–Planck equation for the the distribution function $f_e(\boldsymbol{r}, \boldsymbol{p}, t)$ of the electrons can be written as

$$\frac{\partial f_e}{\partial t} + \boldsymbol{v}\cdot\nabla f_e + \nabla_{\mathbf{p}}\cdot\left(q_e\left(\boldsymbol{E} + \frac{\boldsymbol{v}}{c}\times\boldsymbol{B}\right)f_e\right) = \sum_s C(f_e, f_s), \tag{2.2.1}$$

where $C(f_a, f_b)$ is the collision term giving the rate of change of species a due to collisions with species b. The total collision term can also be written as the divergence of a flux, $\sum_s C(f_e, f_s) = -\nabla_{\mathbf{p}}\cdot\boldsymbol{\Gamma}_c$. The electric and magnetic fields, $\boldsymbol{E}$

and $\boldsymbol{B}$, consist of a fast fluctuating part due to waves injected into or generated inside the plasma and of a part varying slowly in time due to externally applied static fields. By introducing a time averaging over the fast timescale of the fluctuations any variable Q can be separated in a slowly varying part $\langle Q \rangle_t = \hat{Q}$ and a fluctuating part $\tilde{Q}$. After linearization of the fluctuating part of the equation one then obtains the following pair of coupled equations

$$\frac{\partial \hat{f}_e}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} \hat{f}_e + \boldsymbol{\nabla}_{\mathbf{p}} \cdot \left[q_e \left(\hat{\boldsymbol{E}} + \frac{\boldsymbol{v}}{c} \times \hat{\boldsymbol{B}} \right) \hat{f}_e + \boldsymbol{\Gamma}_{ql} \right] = \sum_s \hat{C}(f_e, f_s), \tag{2.2.2}$$

and

$$\frac{\partial \tilde{f}_e}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} \tilde{f}_e + \boldsymbol{\nabla}_{\mathbf{p}} \cdot \left[q_e \left(\hat{\boldsymbol{E}} + \frac{\boldsymbol{v}}{c} \times \hat{\boldsymbol{B}} \right) \tilde{f}_e \right] = -\boldsymbol{\nabla}_{\mathbf{p}} \cdot \left[q_e \left(\tilde{\boldsymbol{E}} + \frac{\boldsymbol{v}}{c} \times \tilde{\boldsymbol{B}} \right) \hat{f}_e \right]. \tag{2.2.3}$$

The time-averaged collective effects of the fluctuating fields are contained in the *quasilinear* flux,

$$\boldsymbol{\Gamma}_{ql} = \left\langle q_e \left(\tilde{\boldsymbol{E}} + \frac{\boldsymbol{v}}{c} \times \tilde{\boldsymbol{B}} \right) \tilde{f}_e \right\rangle_t, \tag{2.2.4}$$

which is second order in the fluctuating fields. In the following it is assumed that the quasilinear flux is known after solution of Eq. (2.2.3).

One can distinguish a hierarchy of timescales in the problem

$$\omega \sim \omega_{ce} \gg \omega_B \gg \nu_c \sim \nu_{ql}, \tag{2.2.5}$$

where ω is the frequency of the fluctuating fields, ω_{ce} is the gyro-frequency and ν_c and ν_{ql} are the time rates of change due to collisions and quasilinear diffusion, respectively. The time average above is now seen to be on a timescale intermediate between the cyclotron or wave period and the bounce-period. The amplitude of the wave fields and of the externally applied static field is now allowed to change on the collisional or quasilinear timescale. We want to solve Eq. (2.2.2) on the slowest (quasilinear/collisional) timescale. This is achieved in the following way by the subsequent averaging over the gyro- and bounce-phases.

To perform the gyro-phase averaging the time-averaged Fokker–Planck equation (2.2.2) is rewritten as

$$\frac{\partial \hat{f}_e}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} \hat{f}_e + \dot{p}\frac{\partial \hat{f}_e}{\partial p} + \dot{\theta}\frac{\partial \hat{f}_e}{\partial \theta} + \dot{\phi}\frac{\partial \hat{f}_e}{\partial \phi} + \boldsymbol{\nabla}_{\mathbf{p}} \cdot \boldsymbol{\Gamma}_{cql} = 0, \tag{2.2.6}$$

where $\boldsymbol{\Gamma}_{cql}$ is used to denote the sum of the collisional and quasilinear fluxes. The time variation along a particle trajectory of the gyro-phase is given by $\dot{\phi} = \omega_{ce} + \mathcal{O}(\omega_{ce}^0)$, while the time variation of the total momentum and the pitch-angle is $\dot{p} \sim \dot{\theta} = \mathcal{O}(\omega_{ce}^0)$. Next, a solution for $\hat{f}_e$ is sought in terms of a series ordered in inverse powers of ω_{ce}; $\hat{f}_e = f + f_1 + f_2 + \cdots$. Substituting this expansion for $\hat{f}$ in Eq. (2.2.6) and collecting the lowest order terms one obtains

$$\omega_{ce}\frac{\partial f}{\partial \phi} = 0, \tag{2.2.7}$$

which is the expected result that to lowest order f is independent of the gyro-phase. To first order one obtains

$$-\omega_{ce}\frac{\partial f_1}{\partial \phi} = \frac{\partial f}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} f + \dot{p}\frac{\partial f}{\partial p} + \dot{\theta}\frac{\partial f}{\partial \theta} + \boldsymbol{\nabla}_{\mathbf{p}} \cdot \boldsymbol{\Gamma}_{cql}. \tag{2.2.8}$$

Note, that f_1 must be periodic in ϕ. This equation can now be averaged over the gyro-phase. In Ref. [3] this is shown to result in the gyro-kinetic equation

$$\frac{\partial f}{\partial t} + v\cos\theta \hat{\boldsymbol{b}} \cdot \boldsymbol{\nabla} f + q_e \hat{\boldsymbol{E}} \cdot \hat{\boldsymbol{b}}\frac{\partial f}{\partial p_{\|}} - \tfrac{1}{2} p \sin\theta (\boldsymbol{\nabla} \cdot \hat{\boldsymbol{b}})\frac{\partial f}{\partial \theta} + \langle \boldsymbol{\Gamma}_{cql} \rangle_{\phi} = 0, \tag{2.2.9}$$

$\hat{\boldsymbol{b}}$ is a unit vector in the direction of the magnetic field. Using Eq. (2.1.5) and that $\hat{\boldsymbol{b}} \cdot \boldsymbol{\nabla} = d/ds$, the second term in the gyro-kinetic equation can be written as

$$v\cos\theta \hat{\boldsymbol{b}} \cdot \boldsymbol{\nabla} f = \omega_B \frac{\partial f}{\partial \phi_B}. \tag{2.2.10}$$

A similar procedure as outlined above can now be used to show that to lowest order in the bounce-period the distribution function is independent of the bounce-phase. The averaging over the bounce-phase removes the second and fourth terms in Eq.

(2.2.9), where the latter term describes the effect of the mirror force. Finally, one then obtains the sought for bounce-averaged Fokker–Planck equation

$$\frac{\partial f_e}{\partial t} = \left\langle \sum_s C(f_e, f_s) \right\rangle_{\phi_B} - \left\langle \left\langle \hat{\boldsymbol{\Gamma}}_{ql} \right\rangle_{\phi} \right\rangle_{\phi_B} - \left\langle q_e \hat{\boldsymbol{E}} \cdot \hat{\boldsymbol{b}} \frac{\partial f_e}{\partial p_\parallel} \right\rangle_{\phi_B} . \tag{2.2.11}$$

Here, f_e is used to denote the bounce-phase independent part of the electron distribution function. The operation of bounce-averaging is defined as

$$\begin{aligned} \langle Q \rangle_{\phi_B} &\equiv \frac{1}{2\pi} \oint Q \, d\phi_B \\ &= \frac{1}{\tau_B} \oint Q \frac{ds}{v \cos\theta}. \end{aligned} \tag{2.2.12}$$

Locally in phase space, the sum of the collisional, quasilinear, and electric field driven fluxes can be written in the form [1,3]

$$\begin{aligned} \left(\frac{\partial f_e}{\partial t} \right)_{cqle} = \Bigg[& \frac{1}{p^2} \frac{\partial}{\partial p} \left(A + B \frac{\partial}{\partial p} + C \frac{\partial}{\partial \theta} \right) \\ & + \frac{1}{p^2 \sin\theta} \frac{\partial}{\partial \theta} \left(D + E \frac{\partial}{\partial p} + F \frac{\partial}{\partial \theta} \right) \Bigg] f_e. \end{aligned} \tag{2.2.13}$$

This is also the form in which the equation is represented inside the code FPPAC. In order to leave this structure intact as much as possible, we want to write the bounce-averaged equation in a conformal way. This can be achieved by writing the equation in terms of the momentum p_0 and pitch-angle θ_0 of the particle at the position of minimum field B_0 along the field line. Note that, because of conservation of energy, $p = p_0$. Defining

$$\alpha^2 = \frac{B}{B_0}, \tag{2.2.14}$$

it follows from the invariance of the magnetic moment μ that

$$\sin\theta = \alpha \sin\theta_0, \tag{2.2.15}$$

and, consequently,

$$\frac{\partial}{\partial \theta} = \frac{\cos\theta}{\alpha \cos\theta_0} \frac{\partial}{\partial \theta_0}. \tag{2.2.16}$$

After substitution of the Eqs. (2.2.15) and (2.2.16) in Eq. (2.2.13) one can easily show that after bounce-averaging the following equation is obtained

$$\lambda\left(\frac{\partial f_0}{\partial t}\right)_{cqle} = \left[\frac{1}{p_0^2}\frac{\partial}{\partial p_0}\left(\mathcal{A}_0 + \mathcal{B}_0\frac{\partial}{\partial p_0} + \mathcal{C}_0\frac{\partial}{\partial \theta_0}\right)\right.$$
$$\left. + \frac{1}{p_0^2 \sin\theta_0}\frac{\partial}{\partial \theta_0}\left(\mathcal{D}_0 + \mathcal{E}_0\frac{\partial}{\partial p_0} + \mathcal{F}_0\frac{\partial}{\partial \theta_0}\right)\right] f_0, \tag{2.2.17}$$

where the coefficients $\mathcal{A}_0$ to $\mathcal{F}_0$ are given by

$$\mathcal{A}_0 = \lambda \langle A \rangle_{\phi_B},$$
$$\mathcal{B}_0 = \lambda \langle B \rangle_{\phi_B},$$
$$\mathcal{C}_0 = \lambda \left\langle \frac{\cos\theta}{\alpha \cos\theta_0} C \right\rangle_{\phi_B},$$
$$\mathcal{D}_0 = \lambda \left\langle \frac{\cos\theta}{\alpha^2 \cos\theta_0} D \right\rangle_{\phi_B},$$
$$\mathcal{E}_0 = \lambda \left\langle \frac{\cos\theta}{\alpha^2 \cos\theta_0} E \right\rangle_{\phi_B},$$
$$\mathcal{F}_0 = \lambda \left\langle \frac{\cos^2\theta}{\alpha^3 \cos^2\theta_0} F \right\rangle_{\phi_B}, \tag{2.2.18}$$

and the quantity

$$\lambda = |v_0 \cos\theta_0| \tau_B. \tag{2.2.19}$$

The contributions to the coefficients due to the presence of a DC electric field take a particularly simple form and are given by

$$\mathcal{A}_{E0} = -s^* q_e p_0^2 \cos\theta_0 E_{\|0}, \tag{2.2.20}$$

and

$$\mathcal{D}_{E0} = s^* q_e p_0 \sin^2\theta_0 E_{\|0}, \tag{2.2.21}$$

where

$$s^* = \left| \oint \alpha^2 ds \right| \tag{2.2.22}$$

is non-zero only for passing particles.

2.3 Approximations to the collision operator

The complete collision operator in the general case of anisotropic distribution functions is rather complicated to calculate and requires large amounts of computing time. For example the complete collision operator as calculated in the original version of FPPAC for the homogeneous magnetic field case is responsible for up to 90% of the required computing time [1]. Moreover, in the inhomogeneous magnetic field case, it is not just this operator that must be calculated, but its bounce-average. In the general form of the operator the bounce-averaging can only be done numerically and, thus, requires an even larger amount of computing time. However, a number of approximate collision operators exists which, depending on the degree of approximation, still contain most or all of the essential physics. As will be shown below, the bounce-average of these operators can often be obtained by multiplication with some constant correction factor depending only on the details of the magnetic equilibrium and the pitch-angle θ_0.

A full discussion of the various approximations to the collision operator in the context of Fokker–Planck codes can be found in Ref. [4]. The present discussion is restricted to the electron collision operator in a two component plasma, i.e. $\sum_{s=e,i} C(f_e, f_s)$.

2.3.1 *The high velocity limit*

The simplest expression for the collision operator is obtained in the limit of high velocities. When the momentum p is much greater than the thermal momentum, $p_{ts} \equiv \sqrt{m_s T_s}$, of the species s, the non-zero terms in the relativistic collision operator are given by

$$A_c^{e/s} = \Gamma^{e/s}\gamma^2 \frac{m_e^2}{m_s}, \tag{2.3.1a}$$

$$B_c^{e/s} = \Gamma^{e/s}\gamma^3 \frac{m_e^3}{m_s^2}\frac{p_{ts}^2}{p}, \tag{2.3.1b}$$

$$F_c^{e/s} = \Gamma^{e/s}\gamma \frac{m_e}{2p}\sin\theta, \tag{2.3.1c}$$

where $\gamma = \sqrt{1 + p^2/m_e^2c^2}$ and $\Gamma^{e/s}$ is defined by

$$\Gamma^{e/s} = \frac{n_s q_e^2 q_s^2 \ln \Lambda^{e/s}}{4\pi\epsilon_0}. \tag{2.3.2}$$

The Coulomb logarithm $\ln \Lambda^{e/s}$ is

$$\ln \Lambda^{e/s} = \ln \left\{ \frac{m_e m_s}{m_e + m_s} \frac{2\alpha c \lambda_D}{e^2} \max \left(\frac{2\bar{E}}{m} \right)_{a,b}^{1/2} \right\} - \frac{1}{2}, \tag{2.3.3}$$

where α is the fine structure constant, λ_D the Debye length, and $\bar{E}$ the mean energy of species a or b. The high velocity limit gives generally a good description of the electron/ion collision term $C(f_e, f_i)$. In fact, usually only the electron/ion pitch-angle scattering term is taken into account, while $A_c^{e/i}$ and $B_c^{e/i}$ are neglected, as they are much smaller than corresponding terms from the electron/electron collisions. For the study of processes in the tail of the electron distribution, the high velocity limit can also be applied to the electron/electron collisions.

It is noted that the high velocity limit operator conserves neither energy nor momentum. Only the density is conserved.

2.3.2 *The linearized collision operator*

In many cases of interest, the electron/electron collisions require a more accurate treatment, in which also the effects of collisions on the thermal part of the distribution function are treated correctly. However, often the deviation from a Maxwellian distribution is small – here small is meant in the sense that the integrated density of the non-Maxwellian part of the distribution is much smaller than the bulk density, while locally in momentum space, at high velocities, the deviation from a Maxwellian may well be large. The electron distribution function is then expanded about a Maxwellian as

$$f_e(\boldsymbol{p}) = f_{em}(p) + f_{e1}(\boldsymbol{p}), \tag{2.3.4}$$

where $f_{em}(p)$ is the relativistic Maxwellian

$$f_{em}(p) = \frac{n_e \mathrm{e}^{-\mu\gamma}}{4\pi m_e^2 c T_e K_2(\mu)}, \tag{2.3.5}$$

with $\mu = m_e c^2/T_e$ and K_n the n^{th}-order modified Bessel function of the second kind. Neglecting terms of order f_{e1}^2, the electron/electron self-collision operator $C(f_e, f_e)$ can then be approximated by the linearized operator

$$C_{lin}^{e/e}\big(f_e(\boldsymbol{p})\big) = C\big(f_e(\boldsymbol{p}), f_{em}(p)\big) + C\big(f_{em}(p), f_e(\boldsymbol{p})\big). \tag{2.3.6}$$

The first term, representing the effect on $f_e(\boldsymbol{p})$ of the collisions off a background Maxwellian population, can be evaluated using the results applying to the case of a general isotropic background. The non-zero terms for the relativistic operator in case of an isotropic background $f_e^{(0)}$ are [4,5]

$$A_c^{e/e} = \frac{4\pi\Gamma^{e/e}}{3n_e}\, p^2 \quad \left(\int_0^p p' f_e^{(0)}(p') \frac{3v_e' - v_e'^3/c^2}{v_e^2}\, dp' + \int_p^\infty p' f_e^{(0)}(p') \frac{2v_e}{c^2}\, dp'\right), \tag{2.3.7a}$$

$$B_c^{e/e} = \frac{4\pi\Gamma^{e/e}}{3n_e}\, p^2 \quad \left(\int_0^p p'^2 f_e^{(0)}(p') \frac{v_e'^2}{v_e^3}\, dp' + \int_p^\infty p' f_e^{(0)}(p') \frac{1}{v_e'}\, dp'\right), \tag{2.3.7b}$$

$$F_c^{e/e} = \frac{4\pi\Gamma^{e/e}}{3n_e}\, \sin\theta \left(\int_0^p p'^2 f_e^{(0)}(p') \frac{3v_e^2 - v_e'^2}{2v_e^3}\, dp' + \int_p^\infty p' f_e^{(0)}(p') \frac{1}{v_e'}\, dp'\right), \tag{2.3.7c}$$

where $v_e = p/\gamma m_e$. The second term in Eq. (2.3.6) represents the effect on the Maxwellian part of the distribution due to collisions with the non-Maxwellian part f_{e1}. To evaluate this term the total distribution function can be expressed as a sum of Legendre harmonics, $f_e(\boldsymbol{p}) = \sum_{l=0}^{\infty} f_e^l(p) P_l(\cos\theta)$, where $P_l(x)$ are the orthonormal set of Legendre polynomials. It is noted, here, that the $f_e^0(p)$ contains all particles and all energy, whereas $f_e^1(p)P_1(\cos\theta)$ contains all macroscopic momentum. Thus, to ensure the conservation of density, energy, and momentum only the contributions coming from the $l = 0, 1$ parts need to be evaluated. The contribution from $l = 0$, $C\big(f_{em}(p), f_e^0(p)\big)$, can again be calculated using Eqs.

(2.3.7) for the case of an isotropic background, while the contribution from $l = 1$, $C\big(f_{em}(p), f_e^1(p)P_1(\cos\theta)\big)$, is given by [4,5]

$$C\big(f_{em}(p), f_e^1(p)\cos\theta\big) = f_{em}(p)\cos\theta\,\frac{4\pi m_e \Gamma^{e/e}}{n_e}\times$$
$$\left\{\frac{f_e^1(p)}{\gamma}\right.$$
$$+\frac{1}{5}\int_0^p dp'\,p'^2 f_e^1(p')\frac{1}{p_{te}^2}\left[\frac{\gamma}{p^2}\frac{p'}{\gamma'^4}\left(\frac{1}{\mu}(4\gamma'^2+6)-\frac{1}{3}(4\gamma'^3-9\gamma')\right)\right.$$
$$\left.+\frac{\gamma^2}{p^2}\frac{p'}{\gamma'^4}\left(\frac{p'^2}{p_{te}^2}\gamma'-\frac{1}{3}(4\gamma'^2+6)\right)\right]$$
$$+\frac{1}{5}\int_p^\infty dp'\,p'^2 f_e^1(p')\frac{1}{p_{te}^2}\left[\frac{\gamma'}{p'^2}\frac{p}{\gamma^4}\left(\frac{1}{\mu}(4\gamma^2+6)-\frac{1}{3}(4\gamma^3-9\gamma)\right)\right.$$
$$\left.\left.+\frac{\gamma'^2}{p'^2}\frac{p}{\gamma^4}\left(\frac{p^2}{p_{te}^2}\gamma-\frac{1}{3}(4\gamma^2+6)\right)\right]\right\}. \quad (2.3.8)$$

2.3.3 *The truncated collision operator*

A particularly useful approximation can be obtained from the linearized collision operator by letting $f_e^0(p) = f_{em}(p)$ in the evaluation of the second term of Eq. (2.3.6). In that way, the truncated collision operator is obtained,

$$C^{e/e}_{trunc}\big(f_e(\boldsymbol{p})\big) = C\big(f_e(\boldsymbol{p}), f_{em}(p)\big) + C\big(f_{em}(p), f_e^1(p)P_1(\cos\theta)\big). \quad (2.3.9)$$

The truncated operator no longer conserves energy, but still conserves density and momentum. This approximation is, in particular, useful for applications like the calculation of current drive efficiency or resistivity. In that case there is no need to provide an energy loss term to prevent an ever increasing energy due to the power absorbed from the waves or gained from the electric field. Here, the energy is lost by collisions on the Maxwellian bulk, whose temperature is kept fixed. Note that this treatment implicitly assumes that the energy loss of the energetic particles due to other processes like, for example, radial diffusion is negligible.

A still further approximation for $C\big(f_e(\boldsymbol{p}), f_{em}(p)\big)$ is possible in cases where the electron temperature is not too high. In that case the non-relativistic approximations can be used for low velocities (up to a few times thermal) to evaluate the

integrals in Eq. (2.3.7), while for higher velocities the results of Eq. (2.3.1) should be recovered. This is achieved by

$$A_c^{e/e} = \Gamma^{e/e} m_e \left[\gamma^2 \operatorname{erf}(u) - u \operatorname{erf}'(u)\right] \tag{2.3.10a}$$

$$B_c^{e/e} = \Gamma^{e/e} m_e \left[\gamma^3 \frac{p_{te}^2}{p} \operatorname{erf}(u) - \frac{p_{te}}{\sqrt{2}} \operatorname{erf}'(u)\right] \tag{2.3.10b}$$

$$F_c^{e/e} = \Gamma^{e/e} m_e \sin\theta \left[\left(\frac{\gamma}{2p} - \frac{\gamma^3 p_{te}^2}{2p^3}\right) \operatorname{erf}(u) + \frac{p_{te}}{2\sqrt{2}p^2} \operatorname{erf}'(u)\right], \tag{2.3.10c}$$

where

$$\operatorname{erf}(u) = \frac{2}{\sqrt{\pi}} \int_0^u \mathrm{e}^{-x^2}\, dx,$$

$$\operatorname{erf}'(u) = \frac{2}{\sqrt{\pi}} \mathrm{e}^{-u^2},$$

$$u = \frac{p}{\sqrt{2} p_{te}}.$$

For $\gamma = 1$, these expressions yield the well-known non-relativistic result for a Maxwellian background [4]. The powers of γ in the terms proportional to $\operatorname{erf}(u)$ have been added to recover the proper high velocity limit (*cf.* Eqs. (2.3.1)).

2.3.4 The bounce-averaging of the collision operator

So far, only the local collision operator has been calculated and the bounce-averaging remains to be done. In almost all approximations treated in the previous subsections, however, nearly all coefficients can rather simply be written in terms of the corresponding coefficients at the position of minimum field. In fact, in all cases A_c and B_c are independent of position, i.e. $A_c = A_{c0}$ and $B_c = B_{c0}$, while with the help of Eq. (2.2.15) F_c can be written as $F_c = \alpha F_{c0}$. According to Eq. (2.2.18) the required bounce-averaged coefficients are then given by

$$\mathcal{A}_{c0} = \lambda A_{c0}, \tag{2.3.11a}$$

$$\mathcal{B}_{c0} = \lambda B_{c0}, \tag{2.3.11b}$$

$$\mathcal{F}_{c0} = \lambda F_{c0} \left\langle \frac{\cos^2\theta}{\alpha^2 \cos^2\theta_0} \right\rangle_{\phi_B} . \tag{2.3.11c}$$

The term that is to be averaged in Eq. (2.3.11c) can be rewritten as

$$\left\langle \frac{\cos^2\theta}{\alpha^2 \cos^2\theta_0} \right\rangle_{\phi_B} = 1 + \frac{1}{\cos^2\theta_0} \left\langle \frac{1}{\alpha^2} - 1 \right\rangle_{\phi_B} \equiv 1 + \langle \Delta \rangle_{\phi_B} . \tag{2.3.12}$$

The remaining part of the linearized and truncated collision operators, that has to be bounce-averaged is the term $C\big(f_{em}(p), f_e^1(p)P_1(\cos\theta)\big)$ responsible for momentum conservation. Since neither $f_{em}(p)$ nor the integral operators in Eq. (2.3.8) are dependent on the pitch-angle, the bounce-averaged operator is given by

$$\big\langle C\big(f_{em}(p), f_e^1(p)P_1(\cos\theta)\big)\big\rangle_{\phi_B} = C\big(f_{em}(p), \big\langle f_e^1(p)P_1(\cos\theta)\big\rangle_{\phi_B}\big). \tag{2.3.13}$$

In order to calculate $\big\langle f_e^1(p)P_1(\cos\theta)\big\rangle_{\phi_B}$, $f_e^1(p)$ is first expressed in terms of $f_{e0}^1(p)$ at the position of minimum B. Substituting θ_0 for θ in the definition for $f_e^1(p)$ and using Eqs. (2.2.15) and (2.2.16), it is shown that

$$\begin{aligned} f_e^1(p) &\equiv \int_o^\pi \sin\theta\, d\theta\, f_e(p,\theta)P_1(\cos\theta) \\ &= \int_o^\pi d\theta_0\, \frac{\alpha^2 \sin\theta_0 \cos\theta_0}{\cos\theta} f_e(p,\theta_0)P_1(\cos\theta) \\ &= \alpha^2\, f_{e0}^1(p). \end{aligned} \tag{2.3.14}$$

It must be noted here, that trapped particles do not contribute to $f_e^1(p)$, so that the borders of the integration domain need not be adjusted, when the integration variable is changed from θ to θ_0. In a more general case the integration domain should be changed to exclude the particles that cannot reach the particular point in space for which the integral is to be evaluated. When the result derived above is combined with Eqs. (2.2.19) and (2.2.22), it is found that

$$\left\langle f_e^1(p)\cos\theta\right\rangle_{\phi_B} = \frac{s^*}{\lambda} f_{e0}^1(p)\cos\theta_0, \tag{2.3.15}$$

and

$$\left\langle C\big(f_{em}(p), f_e^1(p)P_1(\cos\theta)\big)\right\rangle_{\phi_B} = \frac{s^*}{\lambda} C\big(f_{em}(p), f_{e0}^1(p)P_1(\cos\theta_0)\big). \tag{2.3.16}$$

In this form, also the loss of momentum to the trapped particles is included, as s^* is equal to zero in the trapped particle region, i.e. $s^* = 0$ in the trapped particle region reflects the instantaneous loss of the momentum, that is transferred to the trapped part of the background distribution.

As shown above, the bounce-averaging of the approximate collision operators is easily achieved by the multiplication with appropriate constants of the various terms of the collision operator as calculated at the position of minimum magnetic field. These constants λ, s^*, and $\langle\Delta\rangle_{\phi_B}$ need to be calculated only once by the code, because they depend only on the pitch-angle and on the particular magnetic surface at which the Fokker–Planck equation is being solved. Thus, it can be concluded that the bounce-averaged approximate collision operators can be evaluated without a significant increase in the required computing time. Nevertheless, these approximate operators do contain almost all of the essential physics, i.e. the conservation of momentum and/or energy in like particle collisions.

2.4 Explicit results for a large aspect ratio circular tokamak

What remains to be done is the calculation of the constants involved in the bounce-averaging of the collision operator, i.e., λ or, equivalently, the bounce-period τ_B, the pitch-angle scattering correction $\langle\Delta\rangle_{\phi_B}$, and the correction factor s^* in the momentum conservation, and the electric field terms. For general toroidal equilibria this has to be done through numerical integration. Moreover, such general equilibria can only be obtained numerically. For this purpose an interface with the MHD-equilibrium code HELENA [6] is available, which allows the easy calculation of the required integrals and provides all necessary data on the MHD equilibrium

(see Appendix B). In the case of a low β, large aspect ratio tokamak with circular magnetic surfaces, analytic expressions can be obtained. These are presented below.

In a low β tokamak the magnetic field is proportional to the inverse of the major radius, $B \sim R^{-1}$. Hence, the position of minimum magnetic field is located on the outside of the magnetic surface. The poloidal angle ϑ is defined to be zero at that position. For circular magnetic surfaces, the arclength s along a field line can be expressed in terms of the poloidal angle ϑ by

$$\mathrm{d}s = qR_{axis}\,\mathrm{d}\vartheta + \mathcal{O}(\epsilon^2), \tag{2.4.1}$$

where q is the safety factor, R_{axis} the major radius at the axis of the magnetic surface, and ϵ is the inverse aspect ratio, $\epsilon \ll 1$. Further, the quantity $\alpha^2 = B/B_0$ is

$$\alpha^2 = \frac{1+\epsilon}{1+\epsilon\cos\vartheta}. \tag{2.4.2}$$

Combining these two equations with the definition (2.1.4) of the bounce-period and using Eq. (2.2.15), which yields $\cos\theta = \sqrt{1-\alpha^2\sin^2\theta_0}$, the bounce-period is

$$\begin{aligned}\tau_B &= \frac{qR_{axis}}{v}\int_0^{\vartheta_B}\frac{d\vartheta}{\sqrt{1-\frac{1+\epsilon}{1+\epsilon\cos\vartheta}\sin^2\theta_0}} \\ &= \frac{qR_{axis}}{v}\int_0^{\vartheta_B}\frac{d\vartheta\sqrt{1+\epsilon\cos\vartheta}}{\sqrt{1+\epsilon\cos\vartheta-(1+\epsilon)\sin^2\theta_0}},\end{aligned}$$

where ϑ_B is the bounce angle. Now, the calculation presented in Ref. [3] Appendix 3B is followed. Substituting $\cos\vartheta = 1-2\sin^2\frac{1}{2}\vartheta$ gives the result

$$\tau_B = \frac{qR_{axis}}{v\mu_0}\int_0^{\vartheta_B}\frac{d\vartheta\sqrt{1-\mu_T^2\sin^2\frac{1}{2}\vartheta}}{\sqrt{1-(\mu_T/\mu_0)^2\sin^2\frac{1}{2}\vartheta}}, \tag{2.4.3}$$

where $\mu_0 = \cos\theta_0$ and $\mu_T = \cos\theta_{trap}$ is the cosine of the pitch-angle at the boundary between circulating and trapped particles

$$\mu_T = \cos\theta_{trap} = \sqrt{\frac{2\epsilon}{1+\epsilon}}. \tag{2.4.4}$$

The bounce angle is equal to π for circulating particles, $\mu_0^2 > \mu_T^2$, and is given by $\vartheta_B = 2\arcsin \mu_0/\mu_T$ for trapped particles, $\mu_0^2 \leq \mu_T^2$. The numerator in Eq. (2.4.3) is then written in terms of a series expansion and the integration carried out term by term yielding

$$\tau_B = \frac{2qR_{axis}}{v\mu_0} \sum_{m=0}^{\infty} \alpha_m \mu_T^{2m} J_{2m}, \tag{2.4.5}$$

where the coefficients α_m and the functions J_{2m} are determined recursively by

$$\alpha_0 = 1, \quad \alpha_1 = -\tfrac{1}{2}, \ldots, \alpha_m = \frac{2m-3}{2m}\,\alpha_{m-1} \tag{2.4.5a}$$

and

$$J_{2m} = \frac{1}{2m-1}\left((2m-2)\Big(1+\frac{\mu_0^2}{\mu_T^2}\Big)J_{2m-2} - (2m-3)\frac{\mu_0^2}{\mu_T^2}J_{2m-4}\right) \tag{2.4.5b}$$

with

$$J_0 = \begin{cases} \mathrm{K}\Big(\dfrac{\mu_T^2}{\mu_0^2}\Big) & \text{for circulating particles, } \mu_0^2 > \mu_T^2; \\ \dfrac{\mu_0}{\mu_T}\,\mathrm{K}\Big(\dfrac{\mu_0^2}{\mu_T^2}\Big) & \text{for trapped particles, } \mu_0^2 \leq \mu_T^2; \end{cases} \tag{2.4.5c}$$

and

$$J_2 = \begin{cases} \dfrac{\mu_0^2}{\mu_T^2}\Big[\mathrm{K}\Big(\dfrac{\mu_T^2}{\mu_0^2}\Big) - \mathrm{E}\Big(\dfrac{\mu_T^2}{\mu_0^2}\Big)\Big] & \text{for circulating particles, } \mu_0^2 > \mu_T^2; \\ \dfrac{\mu_0}{\mu_T}\Big[\mathrm{K}\Big(\dfrac{\mu_0^2}{\mu_T^2}\Big) - \mathrm{E}\Big(\dfrac{\mu_0^2}{\mu_T^2}\Big)\Big] & \text{for trapped particles, } \mu_0^2 \leq \mu_T^2; \end{cases} \tag{2.4.5d}$$

where K and E are the complete elliptic integrals of the first and second kind, respectively. The correction to the pitch-angle scattering term is calculated in a similar way with the result

$$\langle \Delta \rangle_{\phi_B} = \frac{-2qR_{axis}\mu_T^2}{v\mu_0\tau_B} \sum_{m=0}^{\infty} \alpha_m \mu_T^{2m} J_{2m+2}. \tag{2.4.6}$$

In general, only the first two terms from the expansions in Eqs. (2.4.5) and (2.4.6) will be used yielding correct results up to and including order ϵ. Moreover, going to higher order in ϵ would also require the inclusion of higher order terms in the metric (2.4.1). The correction factor for the momentum conservation term can be calculated directly from its definition and Eq. (2.4.2). This yields the exact result

$$s^* = \begin{cases} \pi q R_{axis} \dfrac{1+\epsilon}{\sqrt{1-\epsilon^2}}, & \text{for circulating particles, } \mu_0^2 > \mu_T^2; \\ 0, & \text{for trapped particles, } \mu_0^2 \leq \mu_T^2. \end{cases} \tag{2.4.7}$$

These analytical expressions are efficiently calculated in the code with the help of simple but highly accurate approximations to the elliptic integrals (see Ref. [7] Eqs. (17.3.34) and (36)).

2.5 Electron Cyclotron waves

Here, the main results concerning the linear and quasilinear theory of Electron Cyclotron (EC) waves are briefly reviewed. For more details the reader is referred to the extensive litterature on this topic [8]. For the parameters of interest for Electron Cyclotron Resonant Heating (ECRH) or Current Drive (ECCD) in tokamaks, the linear theory provides an adequate description of the wave properties. Only for the very high peak power levels as can be achieved by pulsed Free Electron Laser sources, does one expect nonlinear effects to become important [9]. Furthermore, typical wavelengths in the EC frequency range are much smaller than typical lengthscales in the plasma, so that the WKB approximation can be used. The wave properties are then given by the local dispersion relation.

2.5.1. Linear theory and wave properties

In the discussion of the wave dispersion relation, a local, right-handed Carthesian coordinate system is applied with the 3-axis in the direction of the equilibrium magnetic field and the 1-axis along the perpendicular part $k_\perp$ of the wave vector $\boldsymbol{k}$. Normalized momenta $\boldsymbol{x} = \boldsymbol{p}/m_e c$ will be used, while the distribution function

is normalized to give $\int \mathrm{d}^3\boldsymbol{x}\, f(\boldsymbol{x}) = 1$. The wave refractive index $\boldsymbol{N}$ is given by $\boldsymbol{N} = \boldsymbol{k}c/\omega$.

The wave dispersion and other wave properties are obtained from the dispersion equation, which can be written as

$$\boldsymbol{\Lambda}_{ij}\boldsymbol{E}_j \equiv \big(\boldsymbol{N}_i\boldsymbol{N}_j - N^2\boldsymbol{\delta}_{ij} + \boldsymbol{\epsilon}_{ij}\big)\boldsymbol{E}_j = 0, \qquad \text{and} \quad \Lambda \equiv \det(\boldsymbol{\Lambda}_{ij}) = 0, \quad (2.5.1)$$

where $\boldsymbol{\epsilon}_{ij}$ is the dielectric tensor and $\boldsymbol{\delta}_{ij}$ is the identity matrix. The Hermitian part of the dispersion and dielectric tensors describes the wave propagation, while the anti-Hermitian part describes the wave absorption. When the anti-Hermitian part and the wave absorption are small, the wave power flux and wave power density can be expressed as derivatives of the (Hermitian part of the) dispersion equation [10]. The power flux is then given by the derivative with respect to the wave vector $\boldsymbol{k}$

$$\boldsymbol{P}(\boldsymbol{k},\omega) = \frac{-\omega}{8\pi}\frac{\partial}{\partial \boldsymbol{k}}\boldsymbol{E}^* \cdot \boldsymbol{\Lambda}_h \cdot \boldsymbol{E} = \frac{c}{4\pi}\,\mathrm{Re}(\boldsymbol{E}\times\boldsymbol{B}^*) - \frac{\omega}{8\pi}\boldsymbol{E}^* \cdot \frac{\partial \boldsymbol{\epsilon}_h}{\partial \boldsymbol{k}} \cdot \boldsymbol{E}, \quad (2.5.2)$$

where the first part is the electromagnetic Poynting flux

$$\frac{c}{4\pi}\,\mathrm{Re}(\boldsymbol{E}\times\boldsymbol{B}^*) = \frac{c^2}{4\pi\omega}\big[E^2\boldsymbol{k} - \mathrm{Re}\big((\boldsymbol{k}\cdot\boldsymbol{E})\boldsymbol{E}^*\big)\big], \quad (2.5.3)$$

while the second part is known as the sloshing flux. The latter represents the flux of kinetic energy due to the particles moving coherently with the wave. Similarly, the power density in the waves is given by the derivative of the dispersion equation with respect to the frequency

$$U(\boldsymbol{k},\omega) = \frac{1}{8\pi}\boldsymbol{E}^* \cdot \frac{\partial \omega \boldsymbol{\Lambda}_h}{\partial \omega} \cdot \boldsymbol{E} = \frac{1}{8\pi}|\boldsymbol{B}|^2 + \frac{1}{8\pi}\boldsymbol{E}^* \cdot \frac{\partial \omega \boldsymbol{\epsilon}_h}{\partial \omega} \cdot \boldsymbol{E}. \quad (2.5.4)$$

Here, the first term is the magnetic contribution, and the second term contains the electric and kinetic contributions. The ratio of the power flux to the power density defines the group velocity and describes the propagation of the wave through the plasma

$$\frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}t} \equiv \boldsymbol{v}_{\mathrm{group}} = -\frac{\partial\Lambda/\partial\boldsymbol{k}}{\partial\Lambda/\partial\omega}, \quad (2.5.5)$$

and the evolution of the wave vector is given by

$$\frac{\mathrm{d}\boldsymbol{k}}{\mathrm{d}t} = \frac{\partial\Lambda/\partial\boldsymbol{r}}{\partial\Lambda/\partial\omega}. \tag{2.5.6}$$

Finally, the anti-Hermitian part of the dielectric tensor gives the power that is absorbed by the particles as

$$P_{\mathrm{abs}} = \frac{\omega}{4\pi}\boldsymbol{E}^* \cdot \boldsymbol{\varepsilon}_a \cdot \boldsymbol{E}. \tag{2.5.7}$$

Once the dielectric tensor is known, these relations thus completely describe the wave properties, propagation, and absorption.

The dielectric tensor is written in the usual way as an infinite sum over harmonics,

$$\varepsilon(\boldsymbol{N},\omega) = \delta_{ij} - \frac{\omega_p^2}{\omega^2}\sum_{n=-\infty}^{n=+\infty}\int \mathrm{d}^3\boldsymbol{x}\ \frac{S_{ij}^{(n)}}{n\omega_c/\omega + N_\parallel x_\parallel - \gamma} \tag{2.5.8}$$

with

$$S_{ij}^{(n)} \equiv \begin{pmatrix} x_\perp U\frac{n^2\,\mathrm{J}_n^2}{b^2} & -\mathrm{i}x_\perp U\frac{n\,\mathrm{J}_n\,\mathrm{J}_n'}{b} & x_\parallel U\frac{n\,\mathrm{J}_n^2}{b} \\ \mathrm{i}x_\perp U\frac{n\,\mathrm{J}_n\,\mathrm{J}_n'}{b} & x_\perp U(\mathrm{J}_n')^2 & \mathrm{i}x_\parallel U\,\mathrm{J}_n\,\mathrm{J}_n' \\ x_\parallel U\frac{n\,\mathrm{J}_n^2}{b} & -\mathrm{i}x_\parallel U\,\mathrm{J}_n\,\mathrm{J}_n' & x_\perp W\,\mathrm{J}_n^2 \end{pmatrix}.$$

Here, ω_p is the electron plasma frequency, J_n is the Bessel function of order n with argument $b = N_\perp x_\perp \omega/\omega_c$, and J_n' is its derivative. Note that b is the ratio of the electron Larmor radius over the perpendicular wavelength, i.e. $b = k_\perp\rho$. The quantities U and W are functions of the derivatives of the momentum distribution function

$$U \equiv \frac{\partial f}{\partial x_\perp} + \frac{N_\parallel}{\gamma}\left(x_\perp\frac{\partial f}{\partial x_\parallel} - x_\parallel\frac{\partial f}{\partial x_\perp}\right),$$

and

$$W \equiv \frac{\partial f}{\partial x_\parallel} - \frac{n\omega_c}{\gamma\omega x_\perp}\left(x_\perp\frac{\partial f}{\partial x_\parallel} - x_\parallel\frac{\partial f}{\partial x_\perp}\right).$$

The integration over the pole in Eq. (2.5.8) is to be taken over the proper Landau contour, which must pass below the pole in the complex parallel momentum plane. The matrices $S_{ij}^{(n)}$ are Hermitian, so that the principal value contribution from the integration contributes to the Hermitian part and the contribution from the pole gives the anti-Hermitian part of the dielectric tensor.

Away from the resonances, the dielectric tensor reduces to its cold plasma limit which is a sum of the contributions from the $n = -1$, 0, and $+1$ terms

$$\varepsilon_{ij}^{\text{cold}} = \delta_{ij} - \sum_{n=-1}^{+1} \varepsilon_{ij}^{(n)} = \begin{pmatrix} 1 - \frac{\omega_p^2}{\omega^2-\omega_c^2} & \mathrm{i}\frac{\omega_p^2\omega_c}{\omega(\omega^2-\omega_c^2)} & 0 \\ -\mathrm{i}\frac{\omega_p^2\omega_c}{\omega(\omega^2-\omega_c^2)} & 1 - \frac{\omega_p^2}{\omega^2-\omega_c^2} & 0 \\ 0 & 0 & 1 - \frac{\omega_p^2}{\omega^2} \end{pmatrix}. \tag{2.5.9}$$

The separate contributions from $n = -1$, 0, and $+1$ are given by

$$\varepsilon_{ij}^{(-1)} = \frac{\omega_p^2}{2\omega(\omega+\omega_c)} \begin{pmatrix} 1 & \mathrm{i} & 0 \\ -\mathrm{i} & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$\varepsilon_{ij}^{(+1)} = \frac{\omega_p^2}{2\omega(\omega-\omega_c)} \begin{pmatrix} 1 & -\mathrm{i} & 0 \\ \mathrm{i} & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$\varepsilon_{ij}^{(0)} = \frac{\omega_p^2}{\omega} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The trajectory along which the wave propagates through the plasma is described well by the cold plasma dispersion and can be calculated by means of a ray-tracing code, for example the TORAY code [11,12]. The other wave characteristics, in particular the wave polarization, have to be calculated with the correct resonant contributions to the dielectric tensor.

In the Fokker–Planck code, the non-resonant contributions to the dielectric tensor are calculated from the cold plasma approximation, while the resonant contribution can be obtained by numerical integration of Eq. (2.5.8) with the actual distribution function at the position where the wave beam crosses a magnetic surface. In many cases, however, the local distribution function is well approximated

by a Maxwellian distribution, which allows to calculate the dielectric tensor with considerably less computational effort. This approximation gives the wave characteristics with sufficient accuracy for use in the quasilinear diffusion operator.

In general, the Bessel functions in Eq. (2.5.8) are approximated by the first term from their series representation $\mathrm{J}_n(b) = b^n/2^n n!$. In that case the dispersion equation becomes a simple biquadratic equation for $N_\perp$. The two solutions of this equation are known as the Ordinary or O-mode and as the eXtraordinary or X-mode. These modes are characterized by their polarization. For perpendicular propagation, the electric field vector of the O-mode waves is parallel to the magnetic field, while that of the X-mode is perpendicular to the magnetic field. The latter mode is elliptically polarized and has a significant electrostatic contribution around the fundamental resonance, while at higher frequencies the polarization becomes linear again with the electric field vector also perpendicular to the wave vector. The appropriate wave polarization is obtained from Eq. (2.5.1) after solution of the dispersion relation for the relevant mode.

2.5.2 Electron Cyclotron quasilinear diffusion

The Electron Cyclotron (EC) quasilinear diffusion coefficient is calculated by the test-particle approach [13,14]. After bounce-averaging, and in the limit of geometrical optics the result of this approach is formally identical to that of the bounce-averaging of the quasilinear diffusion coefficient in the locally homogeneous, plane wave limit. Here, only the underlying assumptions and the final results will be presented.

In this approach the diffusion coefficient is most conveniently written in terms of invariants of the unperturbed motion. In general, the diffusion coefficient D_{IJ} for two invariants I and J can be written as

$$D_{IJ} \equiv \frac{\langle \Delta I \Delta J \rangle}{2\tau}, \tag{2.5.10}$$

where the average is over all possible orbits with given I and J and over a suitable time τ. When coherence between successive crossings of the particle through the

wave beam is ignored, the average is over orbits crossing the beam while τ becomes the average time between crossings. Because of the large difference between the wave and bounce-frequencies, only a small collisional perturbation of the orbit already destroys the coherence between successive crossings of the beam and the latter approximation is in general well justified. Moreover, because in the electron cyclotron range of frequencies the wave beam is usually well-localized in real space, the time between successive crossings is often much larger than the bounce-period.

The localization of the wave beam in real space also allows a simplification of the calculations by using Taylor expansions of various beam, equilibrium and electron variables around the position of the beam centre. Furthermore, these variations are only accounted for in the resonance function, while all other quantities are evaluated at the beam centre and at the central resonance. The beam is assumed to originate from a monochromatic wave source, so that the wave frequency ω is well-defined and only the wave vector $\boldsymbol{k}$ varies over the beam. The beam power profile is assumed to be Gaussian in both the toroidal and poloidal directions with widths of L_φ and L_ϑ, respectively.

For electron cyclotron waves the diffusion is mainly in the direction of the perpendicular momentum, which is conveniently written as diffusion of the invariant magnetic moment. The following expression for the diffusion coefficient of the magnetic moment $D_{\mu\mu}$ is then obtained

$$D_{\mu\mu} = \frac{\pi e^2}{m_e^2 \omega} \frac{\gamma p_\perp^2}{B^2} |\bar{G}_\perp|^2 \frac{\mathrm{e}^{-(\gamma - n\omega_c/\omega - N_\parallel x_\parallel)^2/\Delta Q}}{\sqrt{\pi \Delta Q}} \times \frac{P_0 \mathrm{e}^{-\int \alpha \mathrm{d}s}}{\Pi \cos\chi} \frac{B}{2\pi \tau_B v_\parallel R B_p}. \tag{2.5.11}$$

This result can easily be compared with the result from the standard quasilinear theory in the homogeneous field, plane wave limit e.g. Ref. [15]. The first few factors are just the same, including the factor $|\bar{G}_\perp|^2$,

$$\bar{G}_\perp = \left(1 - \frac{k_\parallel v_\parallel}{\omega}\right)\left(\mathcal{E}^+ \mathrm{J}_{n+1} + \mathcal{E}^- \mathrm{J}_{n-1}\right) + \frac{v_\parallel n}{v_\perp \gamma \bar{\omega}} \mathcal{E}_\parallel \mathrm{J}_n = \\ = \frac{n\omega_c}{v_\perp \gamma \omega}\left[v_\perp\left(\mathcal{E}^+ \mathrm{J}_{n+1} + \mathcal{E}^- \mathrm{J}_{n-1}\right) + v_\parallel \mathcal{E}_\parallel \mathrm{J}_n\right], \tag{2.5.12}$$

which accounts for the effect of wave polarization and has been normalized to an electric field amplitude $\mathcal{E} = 1$. Here, $\mathcal{E}^{\pm}$ have their usual meaning of $\mathcal{E}^{\pm} = \mathcal{E}_x \pm i\mathcal{E}_y$.

The next term represents the broadened resonant condition, which in the limit of zero broadening $\Delta Q = 0$ reduces to the delta function resonance at

$$\gamma - \frac{n\omega_c}{\omega} - N_{\|} x_{\|} = 0. \tag{2.5.13}$$

The total resonance broadening is the combined effect of the variation of wave and particle variables both along individual particle trajectories and between the different trajectories crossing the beam:

$$\Delta Q = \left[\frac{L_\vartheta \partial}{r\partial\vartheta}(\gamma - n\omega_c/\omega - N_{\|}x_{\|})\right]^2 + \left[\frac{L_\varphi \partial}{R\partial\varphi}(\gamma - n\omega_c/\omega - N_{\|}x_{\|})\right]^2 + \frac{\gamma^2 R^2 \dot{\varphi}^2}{\omega L_\varphi^2}. \tag{2.5.14}$$

This result is identical to what would be obtained from a bounce-averaging of the local delta function resonance, except for the last term which is additional and describes the resonance broadening due to the finite wave-particle interaction time during a beam crossing. The latter, however, is only important in the near field region of the wave antenna, where the geometrical optics approximation breaks down.

Next, the term $P_0 e^{-\int \alpha ds}$ represents the total wave power crossing the flux surface weighted by the factor $1/\Pi \cos\chi$, where Π is the power flux for a normalized electric field vector (Eq. (2.5.2) with $\boldsymbol{E}$ replaced by the unit vector $\mathcal{E}$) and $\cos\chi$ is the cosine of the angle between the direction of wave propagation and the normal of the flux surface. The total injected power is P_0, while the factor $e^{-\int \alpha ds}$, where α is the absorption coefficient, accounts for the power absorbed so far along the beam trajectory s.

The last multiplicative factor, finally, is a division by the effective flux surface area. In the limit of constant B over a flux surface this term becomes exactly equal to the flux surface area. This is easily verified by substituting $\tau_B = 2\pi q R_{axis}/v_{\|}$ and $q = rB/RB_p$.

The diffusion coefficients for other invariants such as, for example, the energy ε, and the cross-diffusion coefficients can be expressed in terms of the diffusion coefficient for the magnetic moment. For this, it is recalled that the parallel and perpendicular resonant wave diffusion are related by [15]

$$\Delta x_{\parallel} = \frac{x_{\perp} N_{\parallel} \omega}{n \omega_c} \Delta x_{\perp}. \tag{2.5.15}$$

This gives the relations

$$D_{\varepsilon\varepsilon} = D_{\mu\mu} B^2 \left(1 + \frac{N_{\parallel} x_{\parallel} \omega}{n \omega_c}\right)^2, \tag{2.5.16a}$$

$$D_{\varepsilon\mu} = D_{\mu\mu} B \left(1 + \frac{N_{\parallel} x_{\parallel} \omega}{n \omega_c}\right). \tag{2.5.16b}$$

Finally, these coefficients are transformed to the coordinate system (p_0, θ_0) used in the Fokker–Planck code. This is a simple coordinate transformation and is effectuated as follows

$$\begin{pmatrix} D_{p_0 p_0} & D_{p_0 \theta_0} \\ D_{\theta_0 p_0} & D_{\theta_0 \theta_0} \end{pmatrix} = \begin{pmatrix} p_{0\mu} & p_{0\varepsilon} \\ \theta_{0\mu} & \theta_{0\varepsilon} \end{pmatrix} \begin{pmatrix} D_{\mu\mu} & D_{\mu\varepsilon} \\ D_{\varepsilon\mu} & D_{\varepsilon\varepsilon} \end{pmatrix} \begin{pmatrix} p_{0\mu} & \theta_{0\mu} \\ p_{0\varepsilon} & \theta_{0\varepsilon} \end{pmatrix}, \tag{2.5.17}$$

where the elements of the transformation matrix are

$$p_{0\mu} = \left.\frac{\partial p_0}{\partial \mu}\right|_{\varepsilon} = 0 \qquad p_{0\varepsilon} = \left.\frac{\partial p_0}{\partial \varepsilon}\right|_{\mu} = \frac{m_e}{p_0}$$

$$\theta_{0\mu} = \left.\frac{\partial \theta_0}{\partial \mu}\right|_{\varepsilon} = \frac{m_e B_0}{p_0^2} \frac{1}{\sin\theta_0 \cos\theta_0} \qquad \theta_{0\varepsilon} = \left.\frac{\partial \theta_0}{\partial \varepsilon}\right|_{\mu} = -\frac{m_e}{p_0^2} \tan\theta_0. \tag{2.5.18}$$

This results in the following contributions to the diffusion coefficients in Eq. (2.2.17) as defined in the code

$$\mathcal{B}_0 = \lambda m^2 B_0^2 D_{\mu\mu} \left(\frac{B}{B_0}\right)^2 \left(1 + \frac{N_{\parallel} x_{\parallel} \omega}{n \omega_c}\right)^2, \tag{2.5.19a}$$

$$\mathcal{C}_0 = \lambda \frac{m^2 B_0^2}{p_0} D_{\mu\mu} \frac{B}{B_0} \left(1 + \frac{N_{\parallel} x_{\parallel} \omega}{n \omega_c}\right) \times \left(\frac{1}{\sin\theta_0 \cos\theta_0} - \frac{B}{B_0} \left(1 + \frac{N_{\parallel} x_{\parallel} \omega}{n \omega_c}\right) \tan\theta_0\right), \tag{2.5.19b}$$

$$\mathcal{E}_0 = C_0 \sin\theta_0, \tag{2.5.19c}$$

$$\mathcal{F}_0 = \lambda \frac{m^2 B_0^2}{p_0^2} D_{\mu\mu} \sin\theta_0 \left(\frac{1}{\sin\theta_0 \cos\theta_0} - \frac{B}{B_0} \left(1 + \frac{N_\parallel x_\parallel \omega}{n\omega_c} \right) \tan\theta_0 \right)^2, \tag{2.5.19d}$$

while the particle wave interaction does not contribute to the convective terms $\mathcal{A}_0$ and $\mathcal{D}_0$.

The absorption from the EC beam is calculated self-consistently from the total absorbed power on a given flux surface. The absorped power is obtained from the flux-surface average of the rate of change of the distribution function due to the wave driven diffusion

$$\begin{aligned} P_{EC} &= \frac{1}{\oint \mathrm{d}s (B_0/B)} \int \mathrm{d}^3\boldsymbol{p}_0 \, (\gamma - 1) m_e c^2 \lambda \left\langle \frac{\partial f}{\partial t}_{EC} \right\rangle_{\phi_B} \\ &= \frac{1}{\oint \mathrm{d}s (B_0/B)} \int \mathrm{d}^3\boldsymbol{p}_0 \, \frac{1}{\gamma m_e p_0} \left(\mathcal{B}_0 \frac{\partial f}{\partial p_0} + \mathcal{C}_0 \frac{\partial f}{\partial \theta_0} \right), \end{aligned} \tag{2.5.20}$$

where the latter result is obtained after a simple integration by parts and the diffusion coefficients are as defined in Eq. (2.5.19). The absorption coefficient α is then given by

$$\mathrm{d}P = \alpha P_0 \mathrm{e}^{-\int \alpha \mathrm{d}s'} \mathrm{d}s = P_{EC} \mathrm{d}V_\psi, \tag{2.5.21}$$

where $\mathrm{d}V_\psi$ is the infinitesimal volume between neighbouring flux surfaces, and $\mathrm{d}s$ is the optical length of the ray path crossing the surface.

The necessary information on the propagation of the wave beam through the plasma is provided by an interface with the TORAY ray-tracing code. Because the wave parameters and the absorption can vary significantly over a single wave beam, each beam can be divided into a number of beamlets for each of which the wave diffusion and wave absorption is treated separately according to the methods described above. Each beamlet in turn can be represented by a number of individual rays, which allow to calculate the spreads of the various wave and equilibrium parameters over the crossing of a beamlet with a flux surface. The details of the interface between the TORAY ray-tracing code and the Fokker–Planck code are given in Appendix C.

3. STRUCTURE OF THE CODE

The original version of FPPAC consists of a driver program, which is specific for the test problems treated in Refs. [1, 2], and a set of subroutines that form the proper core of the Fokker–Planck equation solver. In the present code RELAX the driver program and the related subroutines and common storage are largely new. Most routines belonging to the proper core of FPPAC have been left unchanged, with a few notable exceptions. In particular the subroutines COEF, which calculates the Fokker–Planck coefficients of the collision operator, and PREPKG1, which sets up the boundary conditions, are changed drastically. However, these changes are such that the original code can be reinserted easily. Below, the subroutines belonging to the core of FPPAC will be indicated by an *.

The major change to the structure of the driver program is the possibility to solve the Fokker–Planck equation simultaneously on multiple magnetic surfaces. The code is set up such that the 2-D phase-space meshes on all surfaces are identical.

A flow diagram of the code is sketched in Fig. 1. The main program first calls the subroutine INITIAL in which the non-default input data is read from the namelist 'FPINPUT'. Apart from reading the input data, INITIAL calls a number of subroutines in which, e.g., the phase-space mesh (by XINIT), a number of constants and constant arrays (by XINITL*), the initial distribution functions (by FINIT), and the corresponding densities, energies, and currents (by GNANDE) are calculated. Next, an initial call to the output routine FPOUT is performed. Then, the main loop for time stepping is entered, the size of the coming time step is calculated and the total time is updated accordingly. Thereafter, the inner loop over the magnetic surfaces is entered. Within this loop, the subroutine SETITUP* is called first, which calls the subroutines PREPKG1*, GNANDS*, and GAMMAI*. In PREPKG1* the boundary conditions are set up. GNANDS* copies the densities and energies for that surface into the package arrays, while GAMMAI* calculates the corresponding Coulomb logarithms. Next, the subroutine COEF* is called, in which the Fokker–Planck coefficients for the chosen

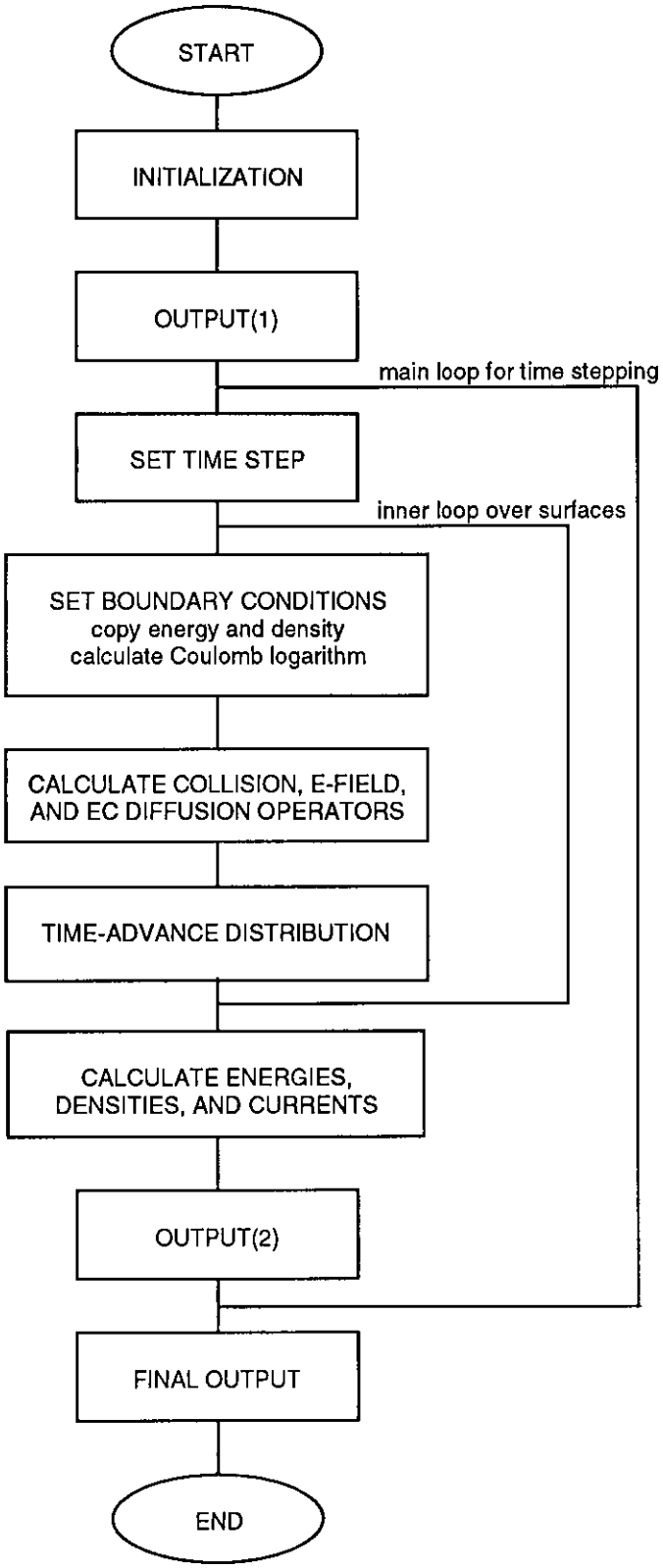


Figure 1. *The general structure of RELAX.*

collision operator are calculated. Subsequently, XSWEEP* is called which performs the actual calculation of the new distribution function. Before the calculation is started, however, the subroutine FPSETUP* is called, which adds in the contributions from additional physics processes to the Fokker–Planck coefficients. These include the contributions to the convective terms due to the DC parallel electric field and the contributions from the EC quasilinear diffusion. In case of fully implicit time stepping, these functions are performed by XSWEEPI*, which then calls a routine to set up the appropriate sparse matrix and then calls a routine to solve the sparse matrix equation (see Appendix A). After the distribution functions are updated for all the surfaces, the densities, energies, and currents associated with the new distributions on each surface are calculated in GNANDE. FPOUT is called to conduct the output that has been requested for that time step. When the requested number of time steps has been performed, a final call to FPOUT is made. The final results for the distribution function are dumped on file for possible future continuation of the calculation or for further analysis by separate post-processor programs.

All calculations associated with the bounce-averaging of the Fokker–Planck equation are performed in the subroutine BOUNCE. An initial call to BOUNCE is made by INITIAL, in order to calculate the constant arrays associated with the bounce-averaging. After the calculation of the collision operator by the subroutine COEF, BOUNCE is called again in order to multiply the collision operator with the appropriate constants (*cf.* Section 2.3.4). Finally, BOUNCE is called a third time after the calculation of the new distribution by XSWEEP, in order to enforce the symmetry of the distribution function in the trapped particle region.

3.1 Spatial and time discretizations

The Fokker–Planck equation is discretized on a momentum/pitch-angle mesh. Both the momentum and pitch-angle meshes are equidistant. The pitch-angle mesh runs from $\theta_0 = 0$ to $\theta_0 = \pi$ with a total of `iy` points. The momentum is normalized to $m_e c$ and the total number of points in the momentum mesh is `jx` ranging from $p = 0$ to $p = p_{\max}$. The letters `i` and `j` will be used to indicate the

pitch-angle and momentum mesh points, respectively.

In the following the subscript 0, indicating the variables at the position of minimum field, is dropped for convenience. The complete Fokker–Planck equation, as it is implemented in the code, is [1,3]

$$Q\left(\frac{\partial f}{\partial t}\right) = \frac{1}{p^2}\frac{\partial}{\partial p}\left(Af + B\frac{\partial f}{\partial p} + C\frac{\partial f}{\partial \theta}\right) + \frac{1}{p^2 \sin\theta}\frac{\partial}{\partial \theta}\left(Df + E\frac{\partial f}{\partial p} + F\frac{\partial f}{\partial \theta}\right) + Kf + S, \tag{3.1.1}$$

where A, B, C, D, E, F, K, Q, and S are arbitrary functions of p and θ. The coefficients A, B, and F contain contributions from the approximate collision operators discussed in Section 2.3. The complete collision operator also contributes non-zero terms to the coefficients C, D, and E. The momentum conservation term of the truncated collision operator (*cf.* Section 2.3.3) contributes to the source term S. In the case of bounce-averaging, the coefficient Q is identified with the quantity λ. In order that Kf and S represent the true source and loss terms, they are also multiplied by λ.

The spatial derivatives are discretized as [1–3]

$$\frac{\partial}{\partial p}(Af)_{i,j} \approx \frac{A_{i,j+1/2} f_{i,j+1/2} - A_{i,j-1/2} f_{i,j-1/2}}{\Delta p_j} = \left[\frac{\delta_{i,j+1/2} A_{i,j} f_{i,j} + \left(1 - \delta_{i,j+1/2}\right) A_{i,j+1} f_{i,j+1}}{\Delta p_j}\right] - \left[\frac{\delta_{i,j-1/2} A_{i,j-1} f_{i,j-1} + \left(1 - \delta_{i,j-1/2}\right) A_{i,j} f_{i,j}}{\Delta p_j}\right], \tag{3.1.2a}$$

$$\frac{\partial}{\partial p}\left(B\frac{\partial f}{\partial p}\right)_{i,j} \approx \frac{1}{\Delta p_j}\left[B_{i,j+1/2}\left(\frac{f_{i,j+1} - f_{i,j}}{\Delta p_{j+1/2}}\right) - B_{i,j-1/2}\left(\frac{f_{i,j} - f_{i,j-1}}{\Delta p_{j-1/2}}\right)\right], \tag{3.1.2b}$$

$$\frac{\partial}{\partial p}\left(C\frac{\partial f}{\partial \theta}\right)_{i,j} \approx \frac{1}{2\Delta p_j}\left[C_{i,j+1}\left(\frac{f_{i+1,j+1} - f_{i-1,j+1}}{2\Delta\theta_i}\right) - C_{i,j-1}\left(\frac{f_{i+1,j-1} - f_{i-1,j-1}}{2\Delta\theta_i}\right)\right], \tag{3.1.3c}$$

where

$$\delta_{i,j+1/2} = \frac{1}{2}\exp\left[-\left(\frac{R_{i,j+1/2}}{R_0}\right)^2\right],$$

$$R_{i,j+1/2} = \frac{\Delta p_{j+1/2}|A_{i,j+1/2}|}{|B_{i,j+1/2}|},$$

and

$$\Delta p_{j\pm1/2} = \pm(p_{j\pm1} - p_j), \qquad \Delta p_j = \frac{1}{2}(p_{j+1} - p_{j-1}),$$

$$\Delta\theta_i = \frac{1}{2}(\theta_{i+1} - \theta_{i-1}), \qquad B_{i,j\pm1/2} = \frac{1}{2}(B_{i,j} + B_{i,j\pm1}).$$

The other terms are discretized analogously. Note, that the value of $\delta_{i,j+1/2}$ determines the weight of central versus upwind differencing. It is determined by the ratio of the cell Reynolds number $R_{i,j+1/2}$ and the parameter R_0. When R_0 is set to ∞, central differencing is recovered, while a very small value for R_0 yields upwind differencing. The authors of FPPAC note that in most cases satisfactory results are obtained with $R_0 = 3.5$ [2]. This way, in cases where advection dominates diffusion, the proper upwind differencing is used, while central differencing is used otherwise.

The time advancement is achieved either by one of two semi-implicit methods, the Alternating Direction Implicit (ADI) or the implicit operator splitting method, or by fully implicit time stepping. The implementation of the fully implicit method is described in Appendix A, while the implementation of the ADI method is described extensively in Ref. [1]. The implicit operator splitting method is very similar to the latter and will be discussed briefly below. Equation (3.1.1) is rewritten as

$$\frac{\partial \lambda f}{\partial t} = \frac{1}{p^2}\frac{\partial \mathcal{G}}{\partial p} + \frac{1}{p^2\sin^2\theta}\frac{\partial \mathcal{H}}{\partial \theta} + \lambda K f + \lambda S, \tag{3.1.4}$$

where

$$\mathcal{G} \equiv Af + B\frac{\partial f}{\partial p} + C\frac{\partial f}{\partial \theta}, \qquad \mathcal{H} \equiv Df + E\frac{\partial f}{\partial p} + F\frac{\partial f}{\partial \theta}.$$

In the first half of time step the equation is solved implicitly in the momentum direction p keeping only half of the source and loss terms and discarding the derivative in the θ direction. This yields

$$\lambda_i f_{i,j}^{n+1/2} = \lambda_i f_{i,j}^{n} + \frac{\Delta t}{v_j^2 \Delta v_j}(\mathcal{G}_{i,j+1/2}^{n+1/2} - \mathcal{G}_{i,j-1/2}^{n+1/2}) + \frac{\Delta t}{2}\lambda_i K_{i,j}^{n} f_{i,j}^{n+1/2} + \frac{\Delta t}{2}\lambda_i S_{i,j}^{n}, \tag{3.1.5}$$

for all interior mesh points, $1 < j < jx$. This is to be supplemented by the appropriate boundary conditions at $p = 0$ and $p = p_{\max}$,i.e., $j = 1$ and $j = jx$, respectively. The cross derivative terms in $\mathcal{G}$ are treated explicitly. Equation (3.1.5) can now be written in the following standard form:

$$-\alpha_{i,j}^{n} f_{i,j+1}^{n+1/2} + \beta_{i,j}^{n} f_{i,j+1}^{n} - \gamma_{i,j}^{n} f_{i,j-1}^{n+1/2} = \delta_{i,j}^{n}, \tag{3.1.6}$$

which can be solved using standard techniques, as given by Richtmyer and Morton [16]. The split in the θ direction is performed analogously.

A further complication occurs, however, because of the presence of the boundary between trapped and passing particles. At this boundary three distinct regions of momentum space are in contact, the co- and counter-passing region and the trapped particle region. A proper treatment of the boundary layer will reflect this contact between all three regions. The problems related to the trapped/passing boundary are treated extensively in Chapter 3 of Ref. [3]. In the present code this problem is treated only approximately by explicitly averaging the distribution function after each time step. In the case of fully explicit time advancement this would yield the same result as the treatment put forward in Ref. [3].

3.1.1 Chebyshev acceleration

In most applications a steady-state solution of the Fokker–Planck equation is searched for. This often requires a large number of time steps, because the size of the time step is limited by problems of numerical stability. One way to reduce the number of time steps, that is required to reach a steady state, is to use a variable time step. The particular method, which is implemented in the code, is known as Chebyshev acceleration [8]. In this method the time step is varied according to a given fixed prescription. Namely, the size of the n^{th} time step, Δt_n is given by

$$\Delta t_n = \frac{2\Delta t_0}{\beta + \alpha - (\beta - \alpha)\cos\left(\frac{[2(n \bmod K)+1]\pi}{K}\right)}, \tag{3.1.7}$$

where α, β, and K are constants with $\alpha < \beta$ and $K =$ integer. For large K this yields a time step that varies between a maximum value somewhat less than $\Delta t_0/\alpha$ and a minimum value close to $\Delta t_0/\beta$. The method then works as follows. For a large time step, the short wavelength eigenmodes of the operator are unstable. On the other hand the small wavelength modes are stable and decay rapidly for small time steps. Thus, the modes that are destabilized during the large time steps, are damped efficiently during the subsequent shorter time steps.

For example, for the default values of the constants, $\alpha = 0.25\times10^{-3}$, $\beta = 5.0$, and $K = 20$, the time step varies between $\Delta t_n = 32.2\Delta t_0$ and $\Delta t_n = 0.201\Delta t_0$. The average time step in this case is $\langle \Delta t_n \rangle_n = 3.97\Delta t_0$. The minimum time step can be chosen to be comfortably small for stability, while the average time step can be up to ten times as large as the maximum allowed time step for stability in the fixed time step scheme.

3.1.2 Run-away boundary conditions

In the presence of large electric fields, the collisional drag on high velocity electrons can become smaller than the acceleration by the electric field. This means that some electrons will run away. In order to be able to describe this problem properly in the code, the boundary conditions at $p = p_{\max}$ have to be changed. At this

boundary, the collisional drag can be calculated from the high velocity limit of the collision operator (2.3.1). Moreover, the momentum diffusion term (2.3.1b) is of order p_{te}^2/p compared with the momentum convection term (2.3.1a) and the pitch-angle diffusion term (2.3.1c). Thus, the momentum diffusion term can be ignored. The remaining equation is

$$\lambda\frac{\partial f}{\partial t} = \frac{1}{p^2}\frac{\partial}{\partial p}(\mathcal{A}_{E0} + \mathcal{A}_{c0})f + \frac{1}{p^2\sin\theta_0}\frac{\partial}{\partial\theta_0}\left(\mathcal{D}_{E0} + \mathcal{F}_{c0}\frac{\partial f}{\partial\theta_0}\right), \tag{3.1.8}$$

where the first term on the right-hand side includes the momentum convection due to the electric field and collisions, respectively, and the second term gives the pitch-angle convection due to the field and the pitch-angle diffusion due to collisions. This equation is purely hyperbolic and needs no boundary condition at $p = p_{\max}$. The various terms and their bounce-averages can be found in Section 2.

In the subroutine PREPKG1, Eq. (3.1.8) is solved at $p = p_{\max}$ in the region where the total flux is directed outwards. The solution is obtained through upwind differencing in the momentum direction and using central differencing in the pitch-angle, as described in Section 3.1. No special treatment of the pitch-angle term is required, because it describes convection and diffusion parallel to the boundary. The time discretization is explicit. This solution is then substituted as the boundary condition for the solution of the equation in the rest of momentum space. In the region, where the total flux is directed inwards, the usual fixed boundary conditions are used.

Particles are lost from the computational domain, because of the total outward flux through the boundary at $p = p_{\max}$, $\Gamma_p(\theta, p_{\max})$. Consequently, the numerical density is not conserved. The particle loss is identified with the run-away rate γ [4],

$$\gamma = \frac{1}{n}\int_{p_{\max}} d\boldsymbol{S}\cdot\boldsymbol{\Gamma} = \frac{2\pi}{n}\int_0^\pi p^2\sin\theta\, d\theta\,\Gamma_p(\theta, p_{\max}). \tag{3.1.9}$$

For large t, a steady state is reached which decays at this run-away rate. Following Ref. [4], the solution of the Fokker–Planck equation is written as

$$f(\boldsymbol{p}, t) = f'(\boldsymbol{p}, t)\,\exp\left(-\int_0^t \gamma(t')\,dt'\right). \tag{3.1.10}$$

Now, f' reaches a true steady state for large t. It is the solution of the normal Fokker–Planck equation plus an additional source term equal to the product of the run-away rate and the distribution function

$$\left(\frac{\partial}{\partial t} - \gamma\right) f' = \ldots . \tag{3.1.11}$$

It is possible to solve for f' instead of f by setting `nlcons` = .`true`. in the input namelist. In that case the run-away rate is calculated in the subroutine PREPKG1, after the new boundary conditions are set. The associated source term is then added to S.

3.2 Input specification

The package FPPAC requires a number of variables to be set by parameter statements at compile time [1]. These parameters are used to set the various arrays to appropriate sizes. Some of these parameters must have a certain fixed value for use with the present code RELAX. The original set of parameters has been extended with an additional parameter, `nsurf`, which specifies the number of magnetic surfaces on which the Fokker–Planck equation is to be solved. A list of the parameters is given in Table I. The parameters and inputs related to the use of the general equilibrium or to the use of the EC diffusion operator are discussed in Appendix B and C, respectively.

TABLE I PARAMETERS OF FOKKER–PLANCK CODE

parameter	description
`nsurf`	the number of magnetic surfaces
`jxa`	the number of momentum mesh points
`iya`	the number of pitch-angle mesh points
`nboa`	the number of general species (`nboa` = 1; electrons)
`meqa`	the number of Maxwellian species (`meqa` = 1; ions)
`ksydma`	= 1: f is not assumed symmetric around $\theta = \pi$
`mxa`	= 0: the electrons are a general species
`nfcga`	= 0: semi-implicit time stepping; = 1 fully implicit

The remaining input variables are set to default values in the subroutine INITIAL. Non-default values are read from the namelist FPINPUT. The following is a complete list of input variables in FPINPUT including a description of their meaning.

Physics input variables

variable	default value	description
`fmass(1)`	9.1066×10^{-28} g	electron mass
`fmass(2)`	1.6726×10^{-24} g	ion mass
`bnumb(k)`	1.0	charge of particle species no. `k` in units of e
`reden(k,is)`	2.0×10^{13} cm^{-3}	density of species no. `k` at surface no. `is`
`tini(k,is)`	1.0 keV	temperature of species no. `k` at surface no. `is`
`epslon(is)`	0.0	inverse aspect ratio ϵ of surface no. `is`
`nltrun`	`.false.`	logical for use of truncated collision operator
`efield(is)`	0.0 Vm^{-1}	parallel electric field at surface no. `is`
`nlrnaw`	`.false.`	logical for use of run-away boundary condition
`nlcons`	`.false.`	logical to set compensation for run-away losses
`nlecrh`	`.true.`	logical to select EC-diffusion operator
`ndispr`	1	frequency of evaluation of EC wave properties
`ncoecd`	1	frequency of calculation of EC-diffusion operator
`nlequi`	`.false.`	logical to select use of general equilibrium
`psisur(is)`	1.0	normalized fluxes of the magnetic surfaces

Calculation control variables

variable	default value	description
`irun`	0	integer for run identification
`nstop`	1	number of time steps
`kspadi`	1	to select implicit operator splitting to select ADI (`kspadi` = 2)
`kdneg`	0	to force non-zero distribution, otherwise `kdneg` = 1
`rz`	3.5	factor for central/up-wind differencing [2]
`vnorm`	3.0×10^{10} cm/s	momentum normalization is m_e `vnorm` (may not be changed!)
`xmax`	1.0	maximum normalized momentum
`tstep`	1.0×10^{-6} s	magnitude of time step
`nlcheb`	`.false.`	logical for choice of Chebyshev acceleration
`chalfa`	0.25×10^{-3}	constant α in Chebyshev acceleration
`chbeta`	5.0	constant β in Chebyshev acceleration
`modulo`	20	constant K in Chebyshev acceleration
`nlresu`	`.false.`	`.true.`: initial distribution read from file

Output control variables

variable	default value	description
`nprint`	1	frequency for printed output
`nplot`	1	frequency for plotted output
`nlprint`	`.true.`	logical to select printing of entire distribution
`nlplot`	`.true.`	logical to select plot output
`nlpln`	`.true.`	logical to select plot of type n = 01 ... 15
`nllog`	`.true.`	for logarithmic scales in plots of f

The various types of plots that can be selected are described in the following subsection.

3.3 Output

All output from the code RELAX is performed by a single subroutine FPOUT. This subroutine is called from three distinctive positions in the main program (see Fig. 1). A first call is made just after initialization of the code. When called this first time, FPOUT prints the complete set of input variables, and a list of the pitch-angle and normalized momentum meshes. In addition, the density, energy, and current of the initial distribution functions are printed. Also the selected plots of the initial distribution functions are made. Next, FPOUT is called at the end of every time step. It then checks whether, according to the specified output frequencies, output should be printed or plotted. Every `nprint` time steps the density, energy, and current of the new distribution functions, and the absorbed EC power are printed. Every `nplot` time steps the selected plots of the new distribution functions are made. Finally, FPOUT is called after the run is completed. At that point a number of plots is made of some characteristics of the distribution function as a function of time.

The following types of plot can be selected:

Type 1: A contour plot of the distribution functions in momentum space. Contours are drawn at $f = \max(f)\exp\left(-\frac{1}{2}(j/2)^2\right), j = 1, ..., n$. For a non-relativistic Maxwellian this gives equidistant contours with a spacing of $\delta p = \frac{1}{2}p_{te}$.

Type 2: A plot of the parallel distribution function, $f_\parallel$, as a function of $p_\parallel|p_\parallel|$. When `nllog` = `.true.`, a plot of $\ln f_\parallel$ is made. The parallel distribution function

is defined by

$$f_{\parallel}(p_{\parallel}) \equiv 2\pi \int_0^{\infty} p_{\perp}\, dp_{\perp}\, f(p_{\parallel}, p_{\perp}). \tag{3.3.1}$$

Type 3: A plot of the perpendicular temperature $T_{\perp}$, which is defined by

$$T_{\perp}(p_{\parallel}) \equiv \frac{2\pi}{f_{\parallel}(p_{\parallel})} \int_0^{\infty} p_{\perp}\, dp_{\perp}\, \frac{p_{\perp}^2}{2m} f(p_{\parallel}, p_{\perp}). \tag{3.3.2}$$

Type 4: A plot of the cuts through the distribution function at five values of the pitch-angle: $\theta = 0, \frac{1}{4}\pi, \frac{1}{2}\pi, \frac{3}{4}\pi, \pi$. When `nllog` = `.true.`, $\ln f$ is plotted.

Type 5: A plot of the density as a function of time.

Type 6: A plot of the temperature as a function of time.

Type 7: A plot of the the current as a function of time.

Type 8: A plot of the electric field driven run-away rate as a function of time.

Type 9: A contour plot of the contribution of EC wave diffusion to $\mathcal{B}_0$.

Type 10: The absorbed EC wave power on each flux surface as a function of time.

Type 11: The absorption coefficient and the transmitted power fraction as a function of the minor radius (for all or a given number of rays to be set in the source code).

Type 12: The relative contribution to the current density as a function of energy.

Type 13: The radial profiles of temperature, density and DC electric field.

Type 14: A plot of the change in the distribution function with respect to the Maxwellian distribution.

Type 15: A plot of the radial current density profile.

4. EXAMPLES

4.1 Plasma conductivity

In most cases that will be presented, an electron temperature $T_e = 1$ keV and a plasma density $n_e = 2 \times 10^{19}$ m^{-3} are assumed. The effective charge of the ions is taken to be $Z_{\text{eff}} = 1$. In the code and in the examples all quantities are unnormalized. Except for the momenta which are normalized to mc. In contrast, many authors use normalized momenta and normalized time, with the momenta normalized to the thermal momentum $p_{te} = \sqrt{m_e T_e}$, and the time normalized to the thermal electron collision time

$$\tau_{te} = \frac{p_{te}^3}{m_e \Gamma^{e/e}}. \tag{4.1.1}$$

For the parameters given above one has

$$\frac{p_{te}}{m_e} = 1.33 \times 10^7 \text{ m/s},$$

$$\tau_{te} = 9.23 \times 10^{-6} \text{ s}.$$

A good test of the truncated collision operator is obtained with the calculation of the plasma conductivity for small electric fields. The result is to be compared with the well-known Spitzer conductivity and its correction for finite aspect ratio. For $Z_{\text{eff}} = 1$, the Spitzer conductivity is given by [18]

$$\sigma_{\text{Sp}} = 2 \frac{n_e e^2 \tau_e}{m_e}, \tag{4.1.2}$$

where the slowing-down time τ_e is

$$\tau_e = 3\sqrt{\frac{\pi}{2}} \tau_{te} = 3.44 \times 10^{11} \frac{T_e^{3/2}}{n_e \ln \Lambda^{e/i}},$$

with the electron temperature, T_e, in eV, and n_e in m^{-3}. The correction of the conductivity due to trapped particle effects has been calculated by Coppi and Sigmar [19] to order ϵ,

$$\sigma = \sigma_{\text{Sp}} \left(1.0 - 1.95\sqrt{\epsilon} + 0.95\epsilon\right). \tag{4.1.3}$$

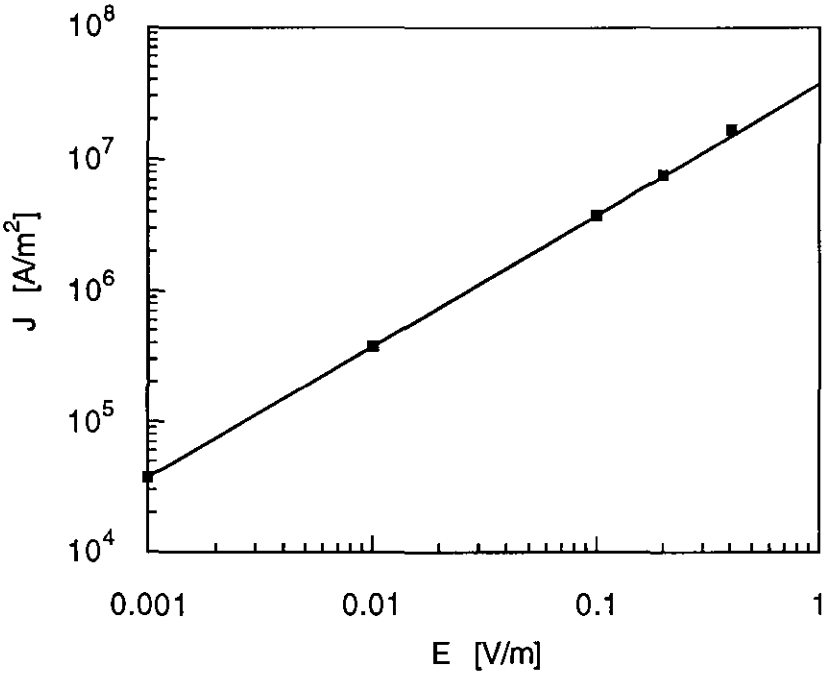


Figure 2. *The current density.*
The current density as a function of the applied electric field E. The truncated collision operator has been used. The results of the code are indicated by the squares. The curve represents Spitzer's resistivity Eq. (4.1.2), see also Table II.

To calculate the conductivity, the code has been run for various values of the electric field. The code is run until a steady state is reached, and the current is calculated. In the calculations the bounce-averaged truncated collision operator is used in its non-relativistic limit. The results are summarized in Figs. 2 and 3 and in Table II. Figure 2 shows the current density as a function of the applied electric field in case of a homogeneous magnetic field. The result clearly shows the linear dependence of the current density on the electric field over several decades. Only towards the high electric fields the run-away regime is entered (*cf.* the next section) and significantly higher current densities are found. In that regime one also finds parts in velocity space where the distribution function becomes negative. This is due to the use of the truncated collision operator, which does not guarantee the non-negativeness of the distribution function.

Figure 3 and Table II give the conductivity as a function of the aspect ratio

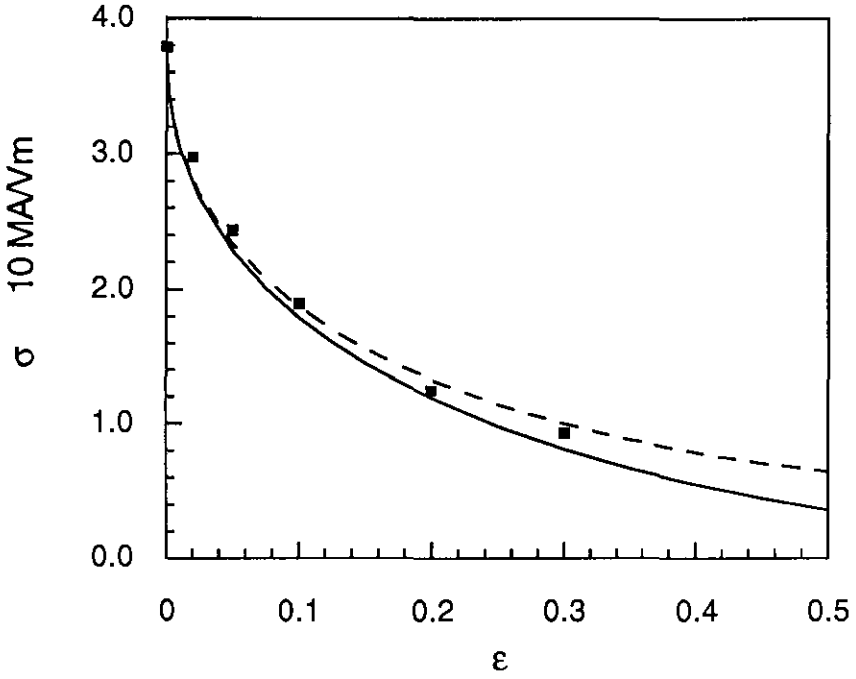


Figure 3. *The conductivity.*
The conductivity as a function of the aspect ratio ϵ. The bounce-averaged, truncated collision operator has been used. The results of the code are indicated by the squares. The curve gives the result of Eq. (4.1.3) and the dashed curve represents a similar expression derived by C.F.F. Karney [20].

TABLE II PLASMA CONDUCTIVITY

collision operator	aspect ratio	conductivity
Spitzer Eq. (4.1.2)		3.75×10^7 A/Vm
truncated	$\epsilon = 0.00$	3.78×10^7 A/Vm
truncated	$\epsilon = 0.02$	3.00×10^7 A/Vm
truncated	$\epsilon = 0.05$	2.44×10^7 A/Vm
truncated	$\epsilon = 0.10$	1.90×10^7 A/Vm
truncated	$\epsilon = 0.20$	1.24×10^7 A/Vm
truncated	$\epsilon = 0.30$	0.93×10^7 A/Vm

ϵ. The results of the code agree well with the analytical expression (4.1.3) derived by Coppi and Sigmar. Also, good agreement is found with a similar expression given by Karney [20], which is based on a fit to numerical results obtained from a solution of the adjoint equation.

The convergence properties of the code results have been analysed for varying

time steps and grid sizes. The results for the homogeneous field case have been obtained using the Chebyshev acceleration method (Section 4.1) with $\Delta t_0 = 2.0 \times 10^{-7}$ s, and a grid (θ, p) of 63 by 127 points. Decreasing the time step or increasing the grid size gives identical results to within 1%. Much smaller time steps are required for the finite aspect ratio cases. The bounce-averaged coefficients have significantly larger values of the derivatives, so that much smaller time steps are required for numerical stability. As the time step is decreased, the solution is seen to converge linearly to the case of zero step size. For a fixed step size the difference with the extrapolated result for zero step size is significantly smaller than for the Chebyshev acceleration method with approximately equal average time step. Apparently the errors created by the larger time steps are not damped sufficiently by the smaller steps, rendering the Chebyshev acceleration method inefficient. A significant improvement in the results is obtained by doubling the number of θ mesh points to 127. In particular the results for small but finite aspect ratio are affected.

4.2 Electron run-away

The implementation of the boundary conditions in the case of large electric fields is illustrated by the following example. The parameters are chosen to be close to the similar case presented in Ref. [4] Section 9.3. For the plasma parameters as given in the previous section, this yields an unnormalized electric field of 0.5 V/m. For such a large electric field, the boundary conditions as discussed in Section 3.1.2 must be applied and the corresponding electron run-away rate can be calculated. To obtain a steady-state solution Eq. (3.1.11) is solved, in which the decrease of the density by the run-away is corrected by an appropriate particle source.

The momentum mesh again extends to $p_{max} = 0.5\, m_e c$. A current density and run-away rate at steady state of $j = 1.35 \times 10^7$ A/m^2 and $\gamma = 5.75$ /s, respectively, are obtained. These results compare well with the corresponding results presented in Ref. [4]. In Fig. 4 the resulting distribution function at steady state is presented. It is also instructive to look at the parallel distribution function, $f_{\|}$ defined by Eq. (3.3.1), and the perpendicular temperature, $T_{\perp}$ Eq. (3.3.2). These are given

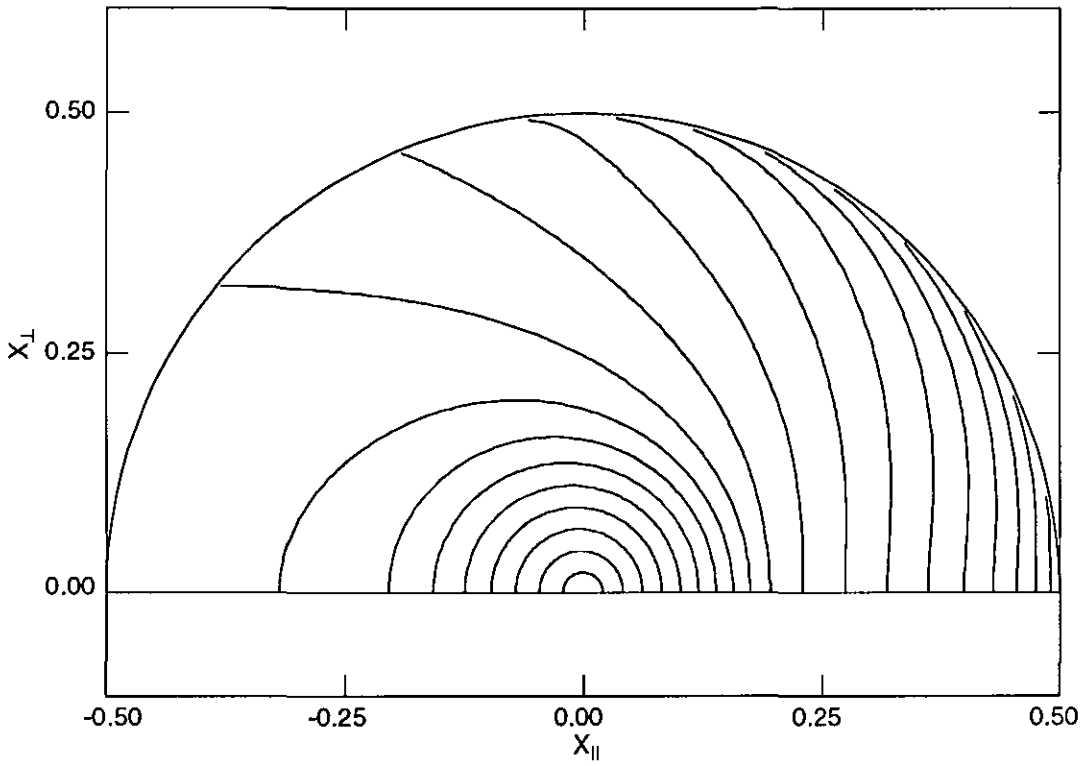


Figure 4. *Steady-state distribution function in the presence of an electric field.* The results of the calculations for $E = 0.5$ V/m, with $p_{max} = 0.5\, m_e c$. Contours of equal phase space density are drawn. The contour levels are proportional to $\exp\left(-\frac{1}{2}(j/2)^2\right)$, $j = 1, ..., n$ (*cf.* Section 3.3).

in Figs. 4 and 5, respectively. For large negative parallel velocities the parallel distribution function becomes almost independent of the velocity. In that region, a strong increase in the perpendicular temperature is found. Because of pitch-angle scattering this also leads to an increase in the perpendicular temperature at positive parallel velocities.

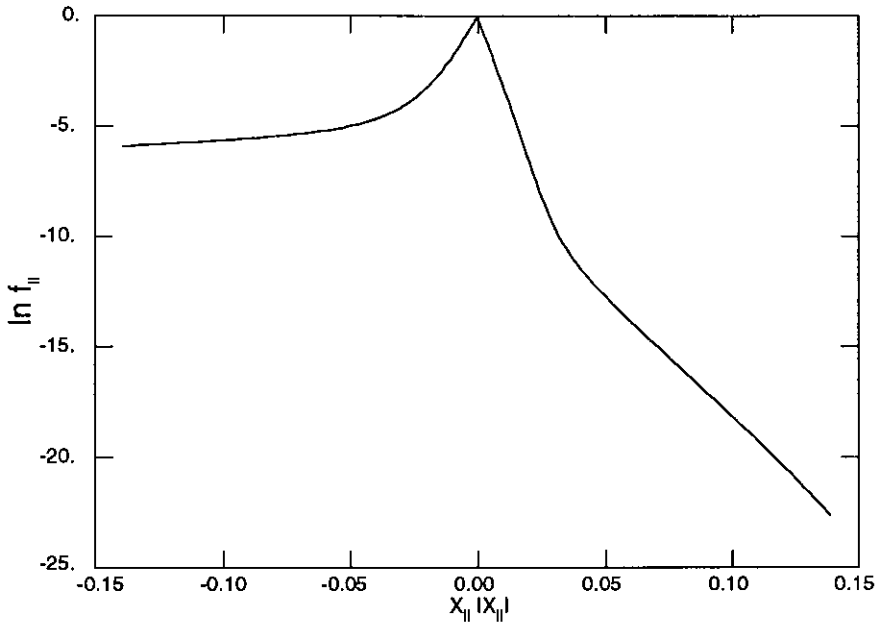


Figure 5. *Parallel distribution function.*

The parallel distribution function corresponding to the case of Fig. 4.

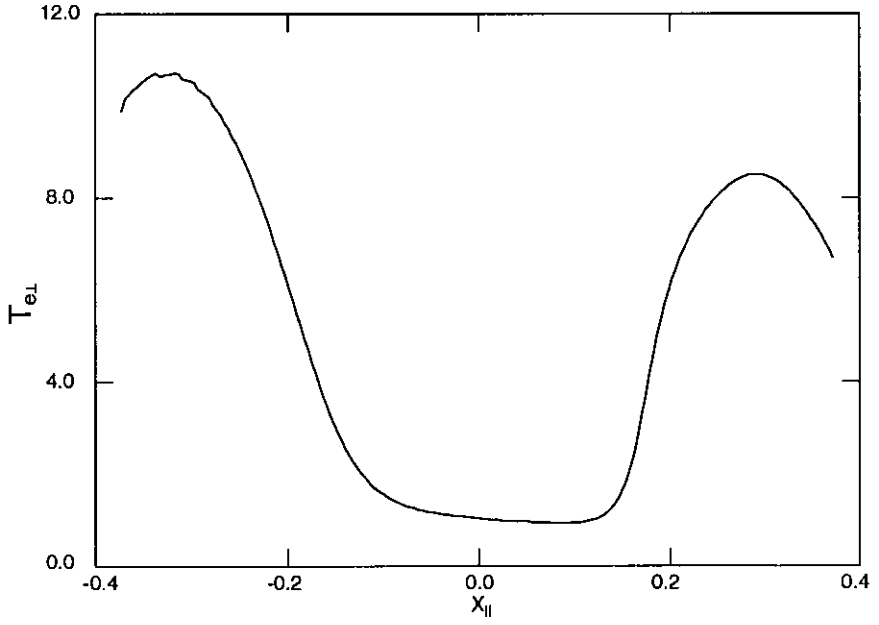


Figure 6. *Perpendicular temperature.*

The perpendicular temperature corresponding to the case of Fig. 4.

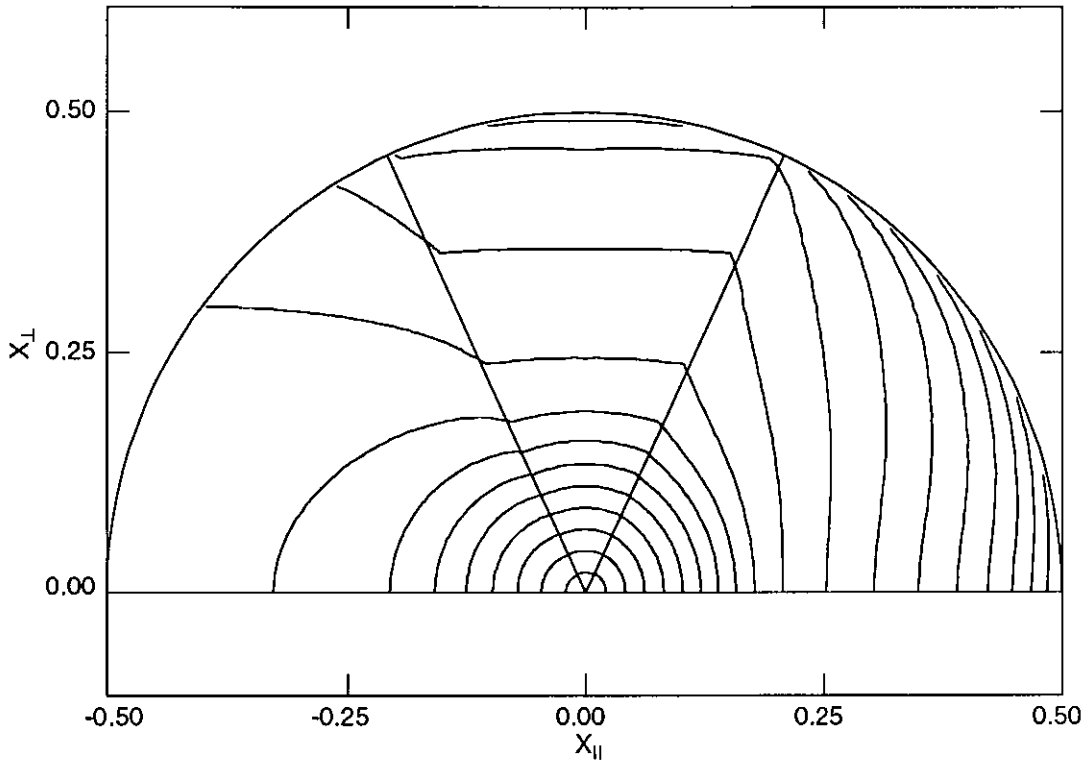


Figure 7. *Steady-state distribution function in the presence of an electric field.* The results of the calculations for finite aspect ratio $\epsilon = 0.1$. The other parameters are as for Fig. 4.

For the same value of the electric field the calculation has been repeated with a finite aspect ratio of $\epsilon = 0.1$. In this case a current density $j = 1.02 \times 10^7$ A/m^2 and run-away rate $\gamma = 5.68$ /s are obtained. Figures 7 to 9 show the properties of the distribution function that is obtained in steady state. The effect of the trapped particle region can be seen clearly. The trapped particle region increases the effectiveness of pitch-angle scattering leading to a significantly higher increase in the parallel distribution for large positive velocities. On the other hand, the trapped particles pin-down the low velocity part of the distribution function more strongly to the thermal distribution. This is most clearly seen on the plot of the perpendicular temperature.

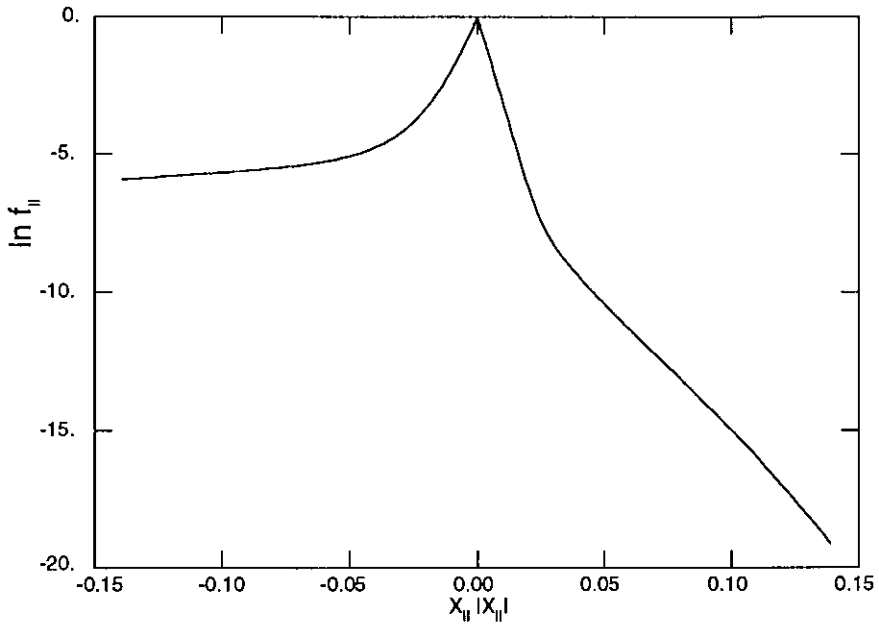


Figure 8. *Parallel distribution function.*

The parallel distribution function corresponding to the case of Fig. 7.

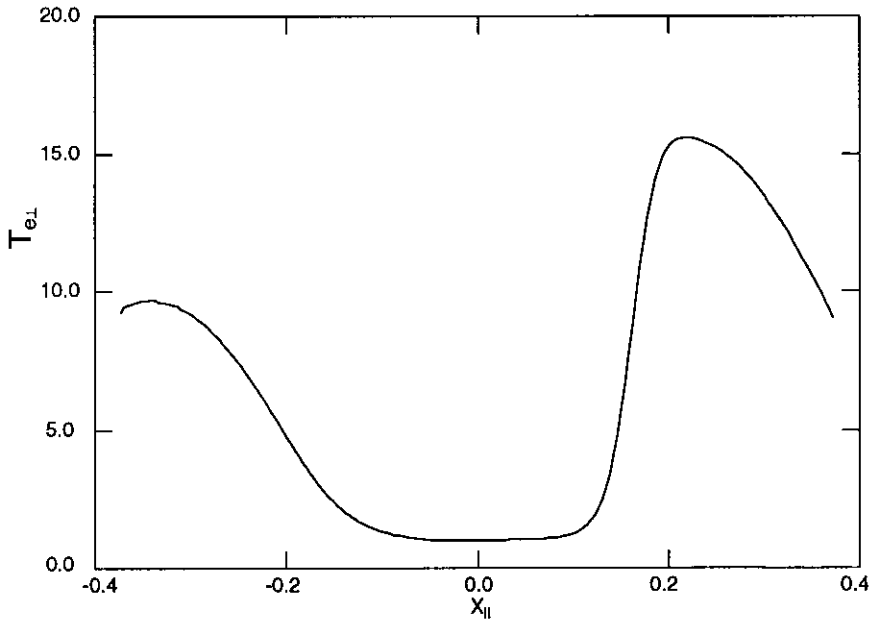


Figure 9. *Perpendicular temperature.*

The perpendicular temperature corresponding to the case of Fig. 7.

4.3 Electron Cyclotron Heating and Current Drive

The EC wave-driven diffusion is illustrated in the following examples of Electron Cyclotron Heating (ECH) and Current Drive (ECCD). These examples also provide important tests of the consistency of the physics models used and on their correct implementation.

The first example is one of ECH in the RTP tokamak ($R_0 = 0.72$m, $a_{\rm lim} = 0.165$m). RTP is equipped with two 60 GHz gyrotrons for the purpose of ECH. One of these is connected by a transmission line to the low field side of the tokamak. The waves are injected perpendicularly to the toroidal field with O-mode polarization. In this example, a case of central resonance is studied, i.e. $B_{\rm axis} = 2.14$ T. For central resonance, the size of the wave beam becomes larger than the poloidal cross section of the relevant magnetic surfaces, which makes this case particularly difficult to treat. The wave beam has to be divided into a large number of rays. The crossings of most surfaces are then again well localized for the individual beamlets, so that the results of Section 2.5.2 remain applicable. Only magnetic surfaces close to the plasma centre have to be included in the calculation, because the wave absorption is well-localized around the resonance.

Figure 10 presents the absorbed power density profile in a thermal plasma, i.e. at the start of the calculation, as calculated from the Fokker–Planck code on a set of 20 equidistant magnetic surfaces covering the central quarter of the plasma, $r_n = (n - \frac{1}{2})\Delta r$ with $\Delta r = .209$ cm. The density and temperature profiles of the plasma are $T_e(r) = 1.5\left[1 - (r/a)^2\right]^2$ keV and $n_e(r) = 2.0\left[1 - (r/a)^2\right] \times 10^{19}$ m^{-3} with constant $Z_{\rm eff} = 1.8$. The total injected power is 120 kW. The EC wave beam is modelled by a rectangular grid in the toroidal and poloidal injection angles of 5×10 beamlets, respectively, while each beamlet, in turn, exists of 10 rays varying only in poloidal injection angle. The information on the rays is obtained from the ray-tracing code TORAY (*cf.* Appendix B). For comparison the power deposition profile calculated by the ray-tracing code is also given. Clearly, the results of both calculations are consistent.

Note, that very high local power densities are achieved in this example, while

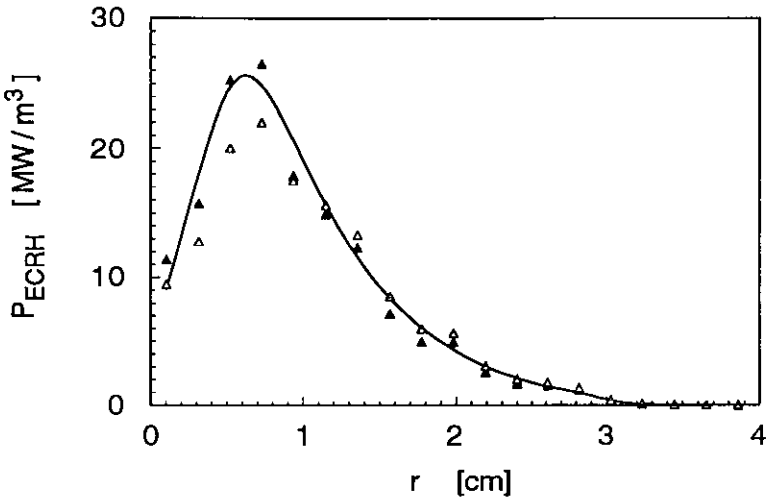


Figure 10. *Electron Cyclotron Heating power deposition profile.*
The power deposition profile for central Electron Cyclotron Heating in RTP (parameters are given in the text). The closed and open triangles give the EC power deposition profile calculated by the Fokker–Planck code for a Maxwellian plasma and for the steady state distribution function obtained at the end of the calculation, respectively. The curve gives the power deposition profile as calculated with the help of the TORAY ray-tracing code for a Maxwellian plasma with the same parameters.

the density is relatively low. Consequently, the wave-driven diffusion will strongly distort the distribution function and create a significant nonthermal electron population. The quasilinear diffusion leads to a flattening of the distribution function in the regions of highest power deposition. This causes a reduction of the power absorption as illustrated in Fig. 10 by the power deposition profile according to the Fokker–Planck calculation, when a steady state is reached. For this case also the soft X-ray emission from the plasma has been calculated, which is sensitive to the amount and energy of the nonthermal electrons. In Fig. 11 this calculated spectrum is compared with an example of a measured spectrum from the RTP experiments [21].

The second example is concerned with non-inductive current drive by EC waves. Here, the efficiency of non-inductive current drive, defined as the ratio of

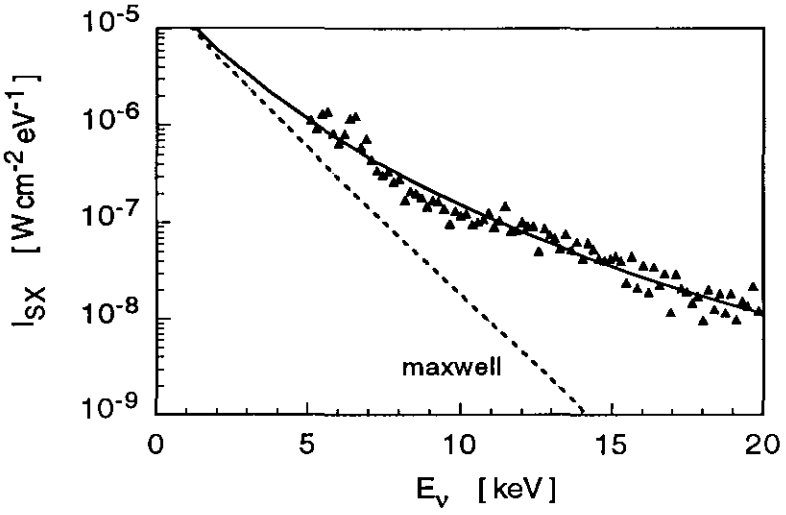


Figure 11. *The soft X-ray spectrum during ECRH.*
The intensity of the soft X-ray emission of the plasma is given as a function of the photon energy. The triangles represent the experimental measurements and the full curve gives the soft X-ray intensity spectrum calculated from the steady-state electron distribution during ECRH predicted by the Fokker–Planck code. For comparison also the spectrum for a Maxwellian plasma is given by the dashed curve.

driven current over the amount of absorbed power, is of particular importance. The current drive efficiency is expected to be strongly affected by trapped-particle effects. These effects have been the subject of various studies, see for example Ref. [22]. In our case the wave and plasma parameters have been chosen to be close to the cases analysed in Ref. [22] by G. Giruzzi. A major radius $R_0 = 2.25$ m is used and the temperature and density are chosen to be $T_e = 2.25$ keV, and $n_e = 4.13 \times 10^{19}$ m^{-3}. The EC waves are injected obliquely from the high field side in X-mode polarization. The parallel refractive index is set to $N_{\|} = 0.75$ with a small (Gaussian) spread of $\Delta N_{\|} = 0.03$, while the ratio of the cyclotron frequency over the wave frequency is $\omega_c/\omega = 1.2$. The latter parameters are supposed to be achieved at either the high field side, $\vartheta = \pi$, or at the low field side $\vartheta = 0$ of the magnetic surface. These two cases are expected to show significant differences with respect to the influence of trapped particles.

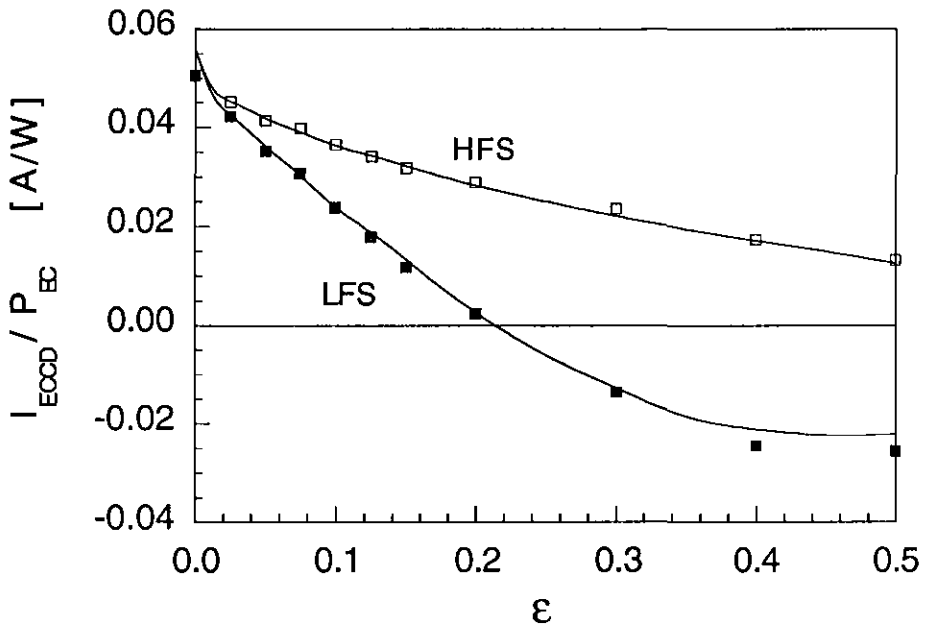


Figure 12. *Efficiency of Electron Cyclotron Current Drive.*
The efficiency of ECCD as a function of the aspect ratio ϵ. The bounce-averaged Maxwellian background collision operator has been used. The results of the code are indicated by the squares (closed and open squares for EC power deposition on the low field side (LFS) and the high field side (HFS), respectively. The curves give the results that are obtained from a calculation with the adjoint method of R.H. Cohen [23].

Figure 12 presents the results in terms of the calculated current drive efficiency for various values of the inverse aspect ratio ϵ. These calculations have been performed at low power, $P_{\mathrm{ECH}} = 1$ kW, using the Maxwellian background collision operator (i.e. without accounting for momentum conservation in the electron-electron collisions). For such a low power, quasilinear modifications of the distribution function are expected to be negligible. This allows a direct comparison of the Fokker–Planck code predictions with results obtained by means of the adjoint method. The curves in Fig. 12 represent the results that are obtained with the adjoint method as presented by R.H. Cohen in Ref. [23]. Clearly, the results obtained by these different methods agree well. Only the reversal of the current for resonance on the low field side and at high values of ϵ appears to be slightly underestimated by the adjoint calculation.

Appendix A. Fully implicit time stepping

In the case of fully implicit time stepping the equations that need to be solved can be cast in the form

$$M_{l,k} f_k^{n+1} = f_k^n, \tag{A.1}$$

where f_k^{n+1} and f_k^n are vectors containing the values of the distribution functions at the new and the old time step, respectively, and $M_{l,k}$ is a sparse, banded matrix resulting from the discretized Fokker–Planck equation. The total size of the matrix is $\mathtt{iy} \times (\mathtt{jx} - 1)$ squared (the values of f at $p_{\max}$ are given by the boundary condition). In the homogeneous field case M is a nine-banded matrix as a consequence of the nine-point difference algorithm. The expressions for the different matrix elements will be given in Section A.1. The trapped-particle region and the trapped/passing boundary lead to additional complications in the equations, which will be discussed in Section A.2.

When fully implicit time stepping is selected, the routine XSWEEPI* is called instead of the routine XSWEEP, which is used for the semi-implicit time stepping. This routine then performs the necessary calls to set up the matrix as described below, and to perform the solution. The calculation of the banded matrix is performed by the routine COIMPL*, which is largely identical to the routine available from the original FPPAC package. The actual solution is performed with a set of routines from the LINPACK library [24]. First, an LU-decomposition of the matrix is calculated by direct Gaussian elimination with partial pivoting by the routine SGBFA. Subsequently, the decomposition is used in routine SGBSL to solve the equation.

A.1 Spatial and time discretizations

Time discretization of the Fokker–Planck equation, using fully implicit time differencing gives

$$\frac{f^{n+1} - f^n}{\Delta t} = \frac{1}{p^2} \frac{\partial}{\partial p} \left(A^n f^{n+1} + B^n \frac{\partial f^{n+1}}{\partial p} + C^n \frac{\partial f^{n+1}}{\partial \theta} \right)$$

$$+\frac{1}{p^2\sin\theta}\frac{\partial}{\partial\theta}\left(D^n f^{n+1}+E^n\frac{\partial f^{n+1}}{\partial p}+F^n\frac{\partial f^{n+1}}{\partial\theta}\right). \qquad (A.2)$$

After substitution of the discretized spatial derivatives as described in Section 3.1, the complete discretized equation is

$$\begin{aligned}
&f_{i,j}^{n+1}-f_{i,j}^{n}= \qquad (A.3)\\
&\quad\frac{\Delta t}{p_j^2\Delta p_j}\left[\left[\delta_{i,j+1/2}^n A_{i,j}^n f_{i,j}^{n+1}+\left(1-\delta_{i,j+1/2}^n\right)A_{i,j+1}^n f_{i,j+1}^{n+1}\right]\right.\\
&\qquad\qquad\left.-\left[\delta_{i,j-1/2}^n A_{i,j-1}^n f_{i,j-1}^{n+1}+\left(1-\delta_{i,j-1/2}^n\right)A_{i,j}^n f_{i,j}^{n+1}\right]\right]\\
&+\frac{\Delta t}{p_j^2\Delta p_j}\left[B_{i,j+1/2}^n\left(\frac{f_{i,j+1}^{n+1}-f_{i,j}^{n+1}}{\Delta p_{j+1/2}}\right)-B_{i,j-1/2}^n\left(\frac{f_{i,j}^{n+1}-f_{i,j-1}^{n+1}}{\Delta p_{j-1/2}}\right)\right]\\
&+\frac{\Delta t}{4p_j^2\Delta p_j\Delta\theta_i}\left[C_{i,j+1}^n\left(f_{i+1,j+1}^{n+1}-f_{i-1,j+1}^{n+1}\right)-C_{i,j-1}^n\left(f_{i+1,j-1}^{n+1}-f_{i-1,j-1}^{n+1}\right)\right]\\
&+\frac{\Delta t}{p_j^2\sin\theta_i\Delta\theta_i}\left[\left[\delta_{i+1/2,j}^n D_{i,j}^n f_{i,j}^{n+1}+\left(1-\delta_{i+1/2,j}^n\right)D_{i+1,j}^n f_{i+1,j}^{n+1}\right]\right.\\
&\qquad\qquad\left.-\left[\delta_{i-1/2,j}^n D_{i-1,j}^n f_{i-1,j}^{n+1}+\left(1-\delta_{i-1/2,j}^n\right)D_{i,j}^n f_{i,j}^{n+1}\right]\right]\\
&+\frac{\Delta t}{4p_j^2\sin\theta_i\Delta\theta_i\Delta p_j}\left[E_{i+1,j}^n\left(f_{i+1,j+1}^{n+1}-f_{i+1,j-1}^{n+1}\right)\right.\\
&\qquad\qquad\left.-E_{i-1,j}^n\left(f_{i-1,j+1}^{n+1}-f_{i-1,j-1}^{n+1}\right)\right]\\
&+\frac{\Delta t}{p_j^2\sin\theta_i\Delta\theta_i}\left[F_{i+1/2,j}^n\left(\frac{f_{i+1,j}^{n+1}-f_{i,j}^{n+1}}{\Delta\theta_{i+1/2}}\right)-F_{i-1/2,j}^n\left(\frac{f_{i,j}^{n+1}-f_{i-1,j}^{n+1}}{\Delta\theta_{i-1/2}}\right)\right].
\end{aligned}$$

Putting f^n on the right-hand side and rearranging the f^{n+1} terms results in the following matrix equation

$$\begin{aligned}
&f_{i,j}^{n}= \qquad (A.4)\\
&\quad M_{(i,j);(i-1,j-1)}f_{i-1,j-1}^{n+1}+M_{(i,j);(i,j-1)}f_{i,j-1}^{n+1}+M_{(i,j);(i+1,j-1)}f_{i+1,j-1}^{n+1}\\
&+M_{(i,j);(i-1,j)}f_{i-1,j}^{n+1}+M_{(i,j);(i,j)}f_{i,j}^{n+1}+M_{(i,j);(i+1,j)}f_{i+1,j}^{n+1}\\
&+M_{(i,j);(i-1,j+1)}f_{i-1,j+1}^{n+1}+M_{(i,j);(i,j+1)}f_{i,j+1}^{n+1}+M_{(i,j);(i+1,j+1)}f_{i+1,j+1}^{n+1},
\end{aligned}$$

where

$$M_{(i,j);(i-1,j-1)}=\frac{\Delta t}{4p_j^2\Delta p_j\Delta\theta_i}\left[-C_{i,j-1}^n\right]+\frac{\Delta t}{4p_j^2\sin\theta_i\Delta\theta_i\Delta p_j}\left[-E_{i-1,j}^n\right]$$

$$M_{(i,j);(i,j-1)} = \frac{\Delta t}{p_j^2 \Delta p_j}\left[\delta^n_{i,j-1/2} A^n_{i,j-1}\right] + \frac{\Delta t}{p_j^2 \Delta p_j}\left[\frac{-B^n_{i,j-1/2}}{\Delta p_{j-1/2}}\right]$$

$$M_{(i,j);(i+1,j-1)} = \frac{\Delta t}{4p_j^2 \Delta p_j \Delta\theta_i}\left[C^n_{i,j-1}\right] + \frac{\Delta t}{4p_j^2 \sin\theta_i \Delta\theta_i \Delta p_j}\left[E^n_{i+1,j}\right]$$

$$M_{(i,j);(i-1,j)} = \frac{\Delta t}{p_j^2 \sin\theta_i \Delta\theta_i}\left[\delta^n_{i-1/2,j} D^n_{i-1,j}\right] + \frac{\Delta t}{p_j^2 \sin\theta_i \Delta\theta_i}\left[\frac{-F^n_{i-1/2,j}}{\Delta\theta_{i-1/2}}\right]$$

$$\begin{aligned} M_{(i,j);(i,j)} = 1 &+ \frac{\Delta t}{p_j^2 \Delta p_j}\left[-\delta^n_{i,j+1/2} A^n_{i,j} + \left(1 - \delta^n_{i,j-1/2}\right) A^n_{i,j}\right] \\ &+ \frac{\Delta t}{p_j^2 \Delta p_j}\left[B^n_{i,j+1/2}\left(\frac{1}{\Delta p_{j+1/2}}\right) + B^n_{i,j-1/2}\left(\frac{1}{\Delta p_{j-1/2}}\right)\right] \\ &+ \frac{\Delta t}{p_j^2 \sin\theta_i \Delta\theta_i}\left[-\delta^n_{i+1/2,j} D^n_{i,j} + \left(1 - \delta^n_{i-1/2,j}\right) D^n_{i,j}\right] \\ &+ \frac{\Delta t}{p_j^2 \sin\theta_i \Delta\theta_i}\left[F^n_{i+1/2,j}\left(\frac{1}{\Delta\theta_{i+1/2}}\right) + F^n_{i-1/2,j}\left(\frac{1}{\Delta\theta_{i-1/2}}\right)\right] \end{aligned}$$

$$M_{(i,j);(i+1,j)} = \frac{\Delta t}{p_j^2 \sin\theta_i \Delta\theta_i}\left[-\left(1 - \delta^n_{i+1/2,j}\right) D^n_{i+1,j}\right] + \frac{\Delta t}{p_j^2 \sin\theta_i \Delta\theta_i}\left[\frac{-F^n_{i+1/2,j}}{\Delta\theta_{i+1/2}}\right]$$

$$M_{(i,j);(i-1,j+1)} = \frac{\Delta t}{4p_j^2 \Delta p_j \Delta\theta_i}\left[C^n_{i,j+1}\right] + \frac{\Delta t}{4p_j^2 \sin\theta_i \Delta\theta_i \Delta p_j}\left[E^n_{i-1,j}\right]$$

$$M_{(i,j);(i,j+1)} = \frac{\Delta t}{p_j^2 \Delta p_j}\left[-\left(1 - \delta^n_{i,j+1/2}\right) A^n_{i,j+1}\right] + \frac{\Delta t}{p_j^2 \Delta p_j}\left[\frac{-B^n_{i,j+1/2}}{\Delta p_{j+1/2}}\right]$$

$$M_{(i,j);(i+1,j+1)} = \frac{\Delta t}{4p_j^2 \Delta p_j \Delta\theta_i}\left[-C^n_{i,j+1}\right] + \frac{\Delta t}{4p_j^2 \sin\theta_i \Delta\theta_i \Delta p_j}\left[-E^n_{i+1,j}\right]. \quad (A.5)$$

The elements of the vectors f and matrix M are indexed by k, which is related to (i,j) by

$$k = (j-1)\mathtt{iy} + i.$$

For points (i,j) for which $2 \le i \le \mathtt{iy}-1$, $2 \le j \le \mathtt{jx}-2$ the coefficients of M are computed from Eqs. (A.5). For points (i,j) for which $i = 1, \mathtt{iy}$ and $j = 1, \mathtt{jx}$ boundary conditions must be applied. Details concerning these boundary conditions are described in Refs. [1,3].

A.2 Treatment of the trapped particle region

In the trapped region, distribution function points located at opposite sides of

the $\theta = \pi/2$ axis and having the same distance to this axis, represent equivalent particles. This means, that not only is the distribution function in the trapped-particle region symmetric around $\theta = \pi/2$, the symmetrically placed points in momentum space within the trapped particle region are actually identical. Thus, the points in the trapped-particle region with $\theta > \pi/2$ can be identified with their symmetrical counterparts at $\theta < \pi/2$, and may be omitted from the calculation.

This introduces two additional boundaries into the problem. One internal boundary at the trapped/passing boundary, and one external boundary at $\theta = \pi/2$. At the latter, the boundary conditions follow directly from the symmetry condition in the trapped-particle region

$$\frac{\partial f}{\partial \theta}(\theta = \pi/2) = 0,$$

and the matrix is easily adjusted accordingly.

The treatment of the trapped/passing boundary is slightly more complicated. Let the co-moving and counter-moving legs of orbits closest to the trapped/passing boundary be identified by the θ-indices `itl` and `itu`, respectively. Then, $f^n_{itl+1,j}$ and $f^n_{itu-1,j}$ are the distribution functions at the boundaries just inside the trapped-particle region, and are thus identical. The distribution funtion at the new time step $n+1$ is now calculated by replacing Eq. (A.4) for the points just inside the trapped/passing boundary by

$$f^{n+1}_{itl+1,j} = \frac{f^{\star n+1}_{itl+1,j} + f^{\star n+1}_{itu-1,j}}{2}, \qquad (A.6)$$

where the $f^{\star n+1}_{i,j}$ represent the right-hand sides of Eq. (A.4) which are modified to account for the proper symmetries in the trapped-particle region. This means that the distribution functions on the counter-passing legs of their orbits are replaced with the proper distribution functions on the co-passing legs. The same is done for the coefficients A through F using the proper symmetries in the trapped particle region. This way, one obtains equations relating $f^{n+1}_{itl+1,j}$ not only to its own nine neighbours, but also to the three functions $f^{n+1}_{itu,j}$ and $f^{n+1}_{itu,j\pm1}$. This leads to three additional bands of non-zero elements in the M matrix. This procedure

is equivalent to stating that the change in the elements at `itl` + 1 is due to half of the fluxes from the opposite sides of the trapped/passing boundary.

In a similar way, the expressions for the $(\texttt{itu}, j)^{\text{th}}$ element are changed by using the symmetry relations to identify $f^{n+1}_{itu-1,j}$ with $f^{n+1}_{itl+1,j}$. This again introduces three additional bands of non-zero elements in the matrix. As a result of these operations, the matrix M is thus expanded to a fifteen banded matrix. Points on the counter-passing legs in the trapped particle region now no longer appear in the right-hand sides of the modified Eq. (A.4) and can indeed be deleted from the computation. After solution for the new distribution function they are then easily obtained from their symmetric counter parts.

Appendix B. Interface with the equilibrium code

When the magnetic surfaces cannot be approximated by concentric circular surfaces, the bounce-time and other relevant integrals over the magnetic surfaces have to be evaluated numerically. This requires a detailed knowledge of the plasma equilibrium, satisfying the force balance equation $\nabla p = \boldsymbol{j} \times \boldsymbol{B}$. In axisymmetric geometry the force balance equation leads to the Grad-Shafranov equation

$$R\frac{\partial}{\partial R}\frac{1}{R}\frac{\partial \psi}{\partial R} + \frac{\partial^2 \psi}{\partial Z^2} = \mu_0 R j_\phi, \tag{B.1}$$

where ψ is the poloidal flux, j_ϕ the toroidal current density, R is the major radial coordinate and Z the vertical coordinate. Here, the toroidal current density can be written as a sum of two flux functions involving the pressure $p(\psi)$ and the diamagnetic function $f(\psi) = RB_\phi$

$$j_\phi = R\frac{\mathrm{d}p}{\mathrm{d}\psi} + \frac{f}{\mu_0 R}\frac{\mathrm{d}f}{\mathrm{d}\psi}. \tag{B.2}$$

The problem now is to solve for $\psi(R, Z)$ given the pressure and current density profiles. Also the shape of the plasma boundary must be specified. In general, the solution for $\psi(R, Z)$ can only be found numerically by means of an MHD equilibrium code. An example of such a code is the program HELENA [6].

The HELENA MHD equilibrium code solves for $\psi(R, Z)$ using a cubic finite element method. The final solution can be given in various forms and coordinate systems, all of which use a discretization in equidistant finite elements in a minor radial and in an angular coordinate (see also Fig. B1). The first of these coordinate systems uses the normal geometrical poloidal angle ϑ and a normalized radial coordinate ρ which is the minor radius r normalized to 1 at the plasma boundary, i.e. $\rho \equiv r/a_{\mathrm{bnd}}(\vartheta)$. Note, that the centre of this coordinate system is the geometrical centre of the shape defined by the plasma boundary. The code will provide the solution of Eq. (B.2) in the form of an array of values of the normalized poloidal flux $\bar{\psi} \equiv (\psi - \psi_0)/(\psi_{\mathrm{lim}} - \psi_0)$ on each of the nodes of the finite elements. As the elements are cubic the solution also gives the radial, angular and the mixed derivatives, $\bar{\psi}_\rho$ $\bar{\psi}_\vartheta$ and $\bar{\psi}_{\rho\vartheta}$.

In the other two coordinate systems used, the centre is that of the actual magnetic axis. For both, the minor radial coordinate is given by the normalized poloidal flux $\bar{\psi}$, while the discretization now is equidistant in the square root of the flux $\sqrt{\bar{\psi}}$. The poloidal angle is either the normal geometric angle ϑ', but now, of course, with respect to the magnetic axis or the poloidal angle in which the field lines appear straight ϑ'', i.e. along the field lines the relation $\mathrm{d}\varphi/\mathrm{d}\vartheta'' = q$ holds. The latter coordinate system is particularly convenient for the evaluation of the bounce and field line integrals that are required, since a simple relation between $\mathrm{d}s$ and $\mathrm{d}\vartheta''$ replacing Eq. (2.4.1) can again be obtained,

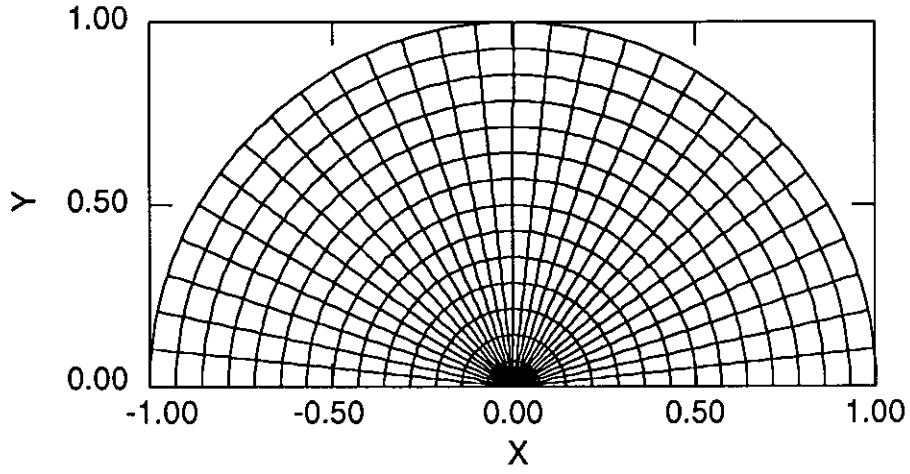
$$\mathrm{d}s = qR\sqrt{1 + (B_p/B_\phi)^2}\ \mathrm{d}\vartheta''. \qquad (B.3)$$

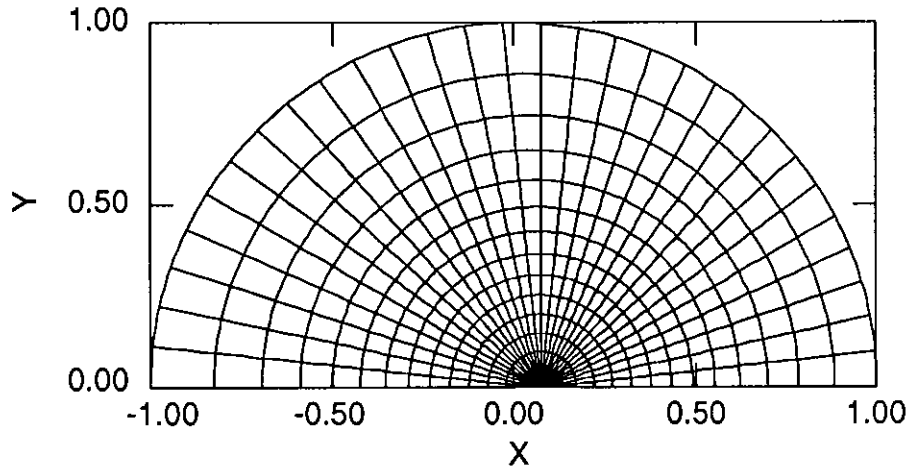
For these two coordinate systems the solution is provided by giving the coordinates R and Z at each of the nodes of the finite elements, while also the relevant derivatives are given again. Given the input profiles and these solutions also allows to calculate the required magnetic fields directly.

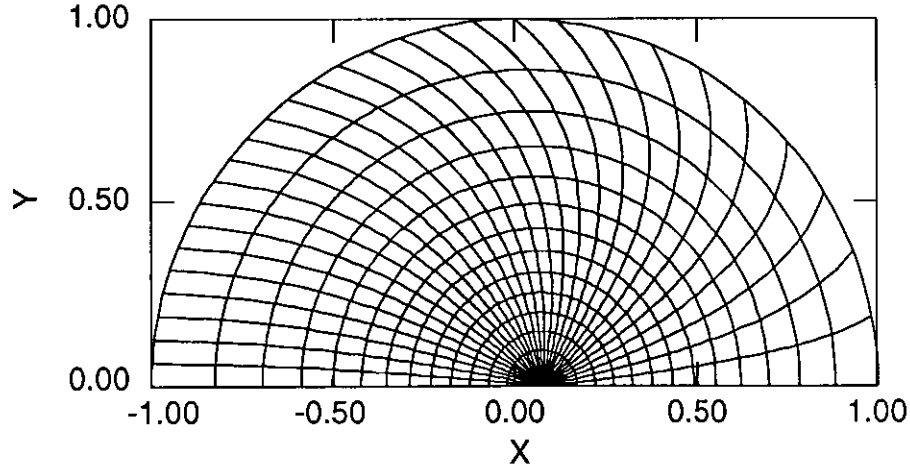
In summary, the HELENA code provides the solution of the Grad-Shafranov equation in the forms $\bar{\psi}(\rho, \vartheta)$, $(R(\bar{\psi}, \vartheta'), Z(\bar{\psi}, \vartheta'))$, and $(R(\bar{\psi}, \vartheta''), Z(\bar{\psi}, \vartheta''))$. The latter of the solutions is written to file and used as input for the Fokker–Planck code in order to perform the field line integrations. The other two forms of the solution are written to file for use with the TORAY ray-tracing code. In this code fast transformations between flux and Carthesian coordinates and vice-versa are required. These are most easily performed with the help of the first two forms of the solution.

Next Page

Figure B1. *The coordinate grids in the MHD equilibrium code HELENA.*
The three coordinate grids that are used by the HELENA MHD code are given for a low β equilibrium with an inverse aspect ratio of $\epsilon = 0.3$ and with a safety factor at the edge of $q_\psi = 5$. From top to bottom the figures give the coordinate lines in the case of the geometrical (ρ, ϑ)-grid, the flux coordinate grid with normal poloidal angle $(\bar{\psi}, \vartheta')$, and the flux coordinate grid with the poloidal angle coordinate in which the field lines appear straight, $\mathrm{d}\phi/\mathrm{d}\vartheta'' = q = \text{constant}$, $(\bar{\psi}, \vartheta'')$, respectively. In the latter two the flux coordinate lines are equidistant in $\sqrt{\bar{\psi}}$.


1.00
0.50
0.00
Y
-1.00
-0.50
0.00
0.50
1.00
X


1.00
0.50
0.00
Y
-1.00
-0.50
0.00
0.50
1.00
X


1.00
0.50
0.00
Y
-1.00
-0.50
0.00
0.50
1.00
X

Appendix C. Interface with the ray-tracing code

The calculation of the EC diffusion operator requires the knowledge of the locations of and wave parameters at the crossings of the wave beam with the magnetic surfaces. Thus, a detailed knowledge of the propagation of the wave beam through the plasma is required. In the limit of geometrical optics, this information can be obtained by solution of the equations for the ray trajectories,

$$\frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}t} \equiv \boldsymbol{v}_{\mathrm{group}} = -\frac{\partial\Lambda/\partial\boldsymbol{k}}{\partial\Lambda/\partial\omega}, \tag{C.1a}$$

and

$$\frac{\mathrm{d}\boldsymbol{k}}{\mathrm{d}t} = \frac{\partial\Lambda/\partial\boldsymbol{r}}{\partial\Lambda/\partial\omega}, \tag{C.1a}$$

where $\Lambda(\omega, \boldsymbol{k}, \boldsymbol{r}, t)$ is the local dispersion relation. In the cold plasma approximation the dispersion equation is of the form

$$\Lambda\big(\omega, \boldsymbol{k}, \boldsymbol{r}, t\big) = \Lambda\big(\omega, \boldsymbol{k}, \omega_p(\boldsymbol{r}, t), \omega_c(\boldsymbol{r}, t)\big). \tag{C.2}$$

Equations (C.1) are solved by the TORAY ray-tracing code for a given set of rays, which together describe the EC wave beam. The information on the rays is written to file for further processing. For the Fokker–Planck code this file is being used to extract the information that is needed.

In the TORAY code the beam is modelled by a set of N_φ by N_ϑ rays lying on a rectangular grid of the toroidal φ and poloidal ϑ injection angles. Each ray is apportioned a part of the beam power in such a way that a Gaussian beam with appropriate spreads is modelled. As mentioned in Section 2.5 the beam is split into a series of beamlets for each of which the quasilinear diffusion coefficients, Eq. (2.5.2) and Eq. (2.5.10), and the absorption coefficient, Eq. (2.5.12), is calculated separately. Each beamlet in turn exists of one or more rays, which are used to evaluate the resonance broadening, ΔQ Eq. (2.5.5), for that beamlet. The power in each beamlet crossing a magnetic surface is equal to the sum of the powers of those rays that actually cross the surface. The wave absorption is

accounted for along each ray-trajectory individually. This means that for each of the rays in a beamlet, the absorption coefficient α is used to calculate the decrease in wave power along its own trajectory.

The necessary information is provided to the Fokker–Planck code as follows. The number of beamlets and rays is set by a series of parameters, while the remaining variables are read from the namelist `raytrece`.

TABLE CI PARAMETERS OF RAY-TRACING

parameter	description
`jhpol`	the number of beamlets in poloidal direction n_ϑ
`ihtor`	the number of beamlets in toroidal direction n_φ
`irdpol`	the number of rays per beamlet poloidally m_ϑ with $m_\vartheta \times n_\vartheta = N_\vartheta$
`irdtor`	the number of rays per beamlet toroidally m_φ with $m_\varphi \times n_\varphi = N_\varphi$
`icocor`	= 1: keep EC operator in core memory, = 0: store on file

TABLE C2 VARIABLES IN NAMELIST `raytrece`

for the entire beam (at each crossing of a magnetic surface):

variable	description
`power`	the total beam power (in erg/s)
`mode`	the mode of the injected waves (X or O)
`nharm`	the harmonic number of the resonance
`ltorex`	the toroidal extension of the entire beam

for all beamlets (at each crossing of a magnetic surface):

variable	description
`nuofcr`	the number of times the surface is crossed (maximum: 2)
`bbo`	the local magnetic field value B/B_0
`omcom`	the local cyclotron frequency ω_c/ω
`domcom`	the spread of the cyclotron frequency $\Delta_\vartheta \omega_c/\omega$
`enpar`	the parallel refractive index $N_\parallel$
`dnparp`	the poloidal spread $\Delta_\vartheta N_\parallel$
`dnpart`	the toroidal spread $\Delta_\varphi N_\parallel$

for all rays (at each crossing of a magnetic surface):

variable	description
`poweri`	the fraction of the total power carried by the ray
`nurrcr`	the number of times the ray crosses the surface (maximum: 2)
`optisc`	$1/\cos\chi \equiv \mathrm{d}s/\mathrm{d}r$
`sdel`	the arclength to the previous surface crossing

Appendix D. Simulation of plasma diagnostics

For a direct comparison of code predictions with experimental measurements the expected diagnostic signals must be calculated on the basis of the electron distribution function calculated by the Fokker–Planck code. In particular, the soft X-ray and Electron Cyclotron Emission (ECE) diagnostics are very sensitive to deviations of the electron distribution function from thermal. The calculation of these two diagnostic signals is achieved by two programs used for post-processing of the Fokker–Planck code results.

The ECE emission is calculated with the help of a modified version of the NOTEC ray-tracing and ECE code [25]. The NOTEC code solves the equation for radiative transfer, including spontaneous emission, along the ray traces. The ray-tracing is again performed with the help of the dispersion equation from cold-plasma theory (see Section 2.5.1). The equation for radiative transfer is [10]

$$N_r^2 \frac{\mathrm{d}}{\mathrm{d}s} \frac{I_\omega}{N_r^2} = \eta_\omega - \alpha_\omega I_\omega, \tag{D.1}$$

where I_ω is the intensity of radiation at the frequency ω, α the absorption coefficient, η the emission coefficient, and N_r the ray refractive index [10]. The absorption coefficient is obtained from the linear theory (combining Eqs. (2.5.2) and (2.5.7))

$$\alpha = \frac{P_{\mathrm{abs}}}{|\boldsymbol{P}|} = \frac{\omega}{4\pi} \frac{\boldsymbol{E}^* \cdot \boldsymbol{\varepsilon}_a \cdot \boldsymbol{E}}{|\boldsymbol{P}|}, \tag{D.2}$$

while the equation for the emission coefficient is obtained from the the fluctuation-dissipation theorem [26] (for a definition of the symbols see Section 2.5.)

$$\eta = N_r^2 \frac{\omega_p^2 \omega m_e}{32\pi^3} \frac{1}{\Pi} \sum_n \int \frac{\mathrm{d}^3 \boldsymbol{p}}{\gamma} f(\boldsymbol{p}) |\mathcal{E} \cdot \boldsymbol{j}_n^*|^2 \delta\big(\gamma - N_\parallel x_\parallel - n\omega_c/\omega\big). \tag{D.3}$$

The electron distribution functions obtained from the Fokker–Planck code are used as input to the NOTEC ray-tracing code for the numerical evaluation of Eqs. (D.1–3). In those regions of the plasma where no data from the Fokker–Planck code are

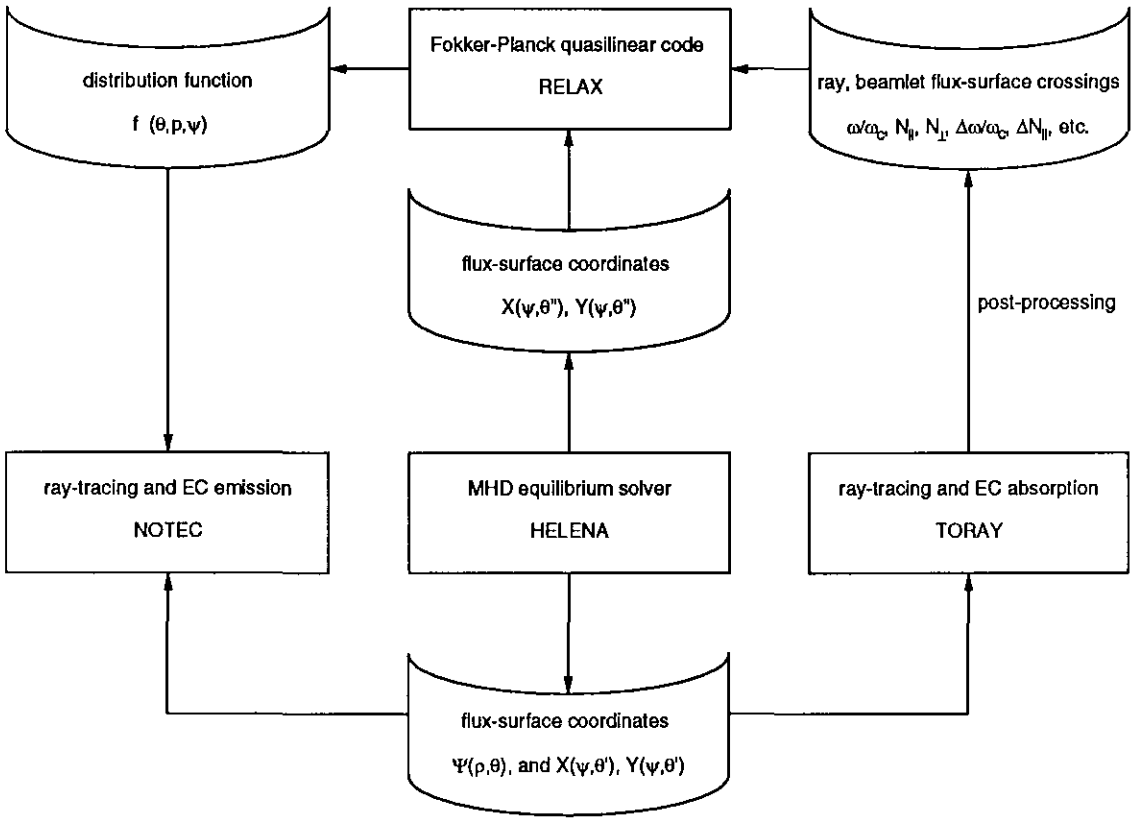


Figure D1. *The relations between the codes.*
The interdependencies of the various computer codes is sketched. The arrows indicate the production or requirement of the data files by the codes.

available the usual relations for a thermal plasma are used. The final results of the NOTEC code are expressed in terms of an equivalent black-body temperature that would correspond to the calculated intensity of the emission at a given wavelength.

Like the TORAY ray-tracing code, NOTEC has also been changed to accept input from the HELENA equilibrium code. An overview of the codes and their interdependencies is sketched in Fig. D1.

A post-processor program has also been written to evaluate the soft X-ray spectrum as obtained from a pin-hole camera. The intensity observed by such a camera is a line integral of the soft X-ray emissivity along the line of sight defined by the pin-hole. For not too high energies of the electrons, the soft X-ray emissivity

is isotropic and is given by

$$I(\varepsilon) = 9.40 \times 10^{-32} Z_{\text{eff}} n_e^2 \int_{\varepsilon}^{\infty} \frac{g(\varepsilon_e)}{\sqrt{\varepsilon_e}} \mathrm{d}\varepsilon_e, \qquad (D.4)$$

where $I(\varepsilon)$ is the total energy radiated per second in units of eV/s by a plasma volume of 1 m^{-3} in the spectral range of $\hbar\nu = (\varepsilon, \varepsilon + \mathrm{d}\varepsilon)$ with ε in eV, n_e the electron density in units of m^{-3}, and g the normalized energy distribution function

$$g(\varepsilon)\,\mathrm{d}\varepsilon = 2\pi\, p^2\, \mathrm{d}p(\varepsilon) \int \mathrm{d}\theta f(p).$$

The integral over the distribution function is performed numerically, using the distribution functions resulting from the Fokker–Planck code. For those parts of the plasma where no Fokker–Planck results are available a Maxwellian distribution is again assumed.

REFERENCES

[1] McCoy, M.G., Mirin, A.A., and Killeen, J., *Comput. Phys. Commun.* **24** (1981) 37.

[2] McCoy, M.G., Mirin, A.A., Tomaschke, G.P., and Killeen, J., *Comput. Phys. Commun.* **51** (1988) 373.

[3] Killeen, J., Kerbel, G.D., McCoy, M.G., and Mirin, A.A., *Computational Methods for Kinetic Models of Magnetically Confined Plasmas*, Springer Verlag, New York, USA (1986).

[4] Karney, C.F.F., *Fokker–Planck and Quasilinear Codes*, proceedings of the 3rd European Workshop on the Problems in the numerical Modeling of Plasmas, Varenna, Italy, September 10–13, 1985, *Computer Physics Reports* 4 (1986) 183.

[5] Karney, C.F.F., and Fisch, N.J., *Phys. Fluids* **28** (1985) 116.

[6] Huysmans, G.T.A., Goedbloed, J.P., and Kerner, W., proceedings of the CP90 Europhysics Conference on *Computational Physics*, 10–13 September 1990, Amsterdam, The Netherlands, Editor A. Tenner, World Scientific, Singapore, (1991) p. 371.

[7] Abramowitz, M., and Stegun, I.A., *Handbook of Mathematical Functions*, National Bureau of Standards, Washington D.C., USA (1965).

[8] Bornatici, M., Cano, R., De Barbieri, O., and Engelmann, F., *Nucl. Fusion* **23** (1983) 1153.

[9] Cohen, B.I., Cohen, R.H., Nevins, W.M., and Rognlien, T.D., *Rev. Mod. Phys.* **63** (1991) 949.

[10] Bekefi, G., *Radiation Processes in Plasmas*, John Wiley and Sons, Inc., New York, USA (1966).

[11] Kritz, A.H., Hsuan, H., Goldfinger, R.C., and Batchelor, D.B., proceedings of the 3rd Intern. Symposium on Heating in Toroidal Plasmas, Grenoble (France), 22–26 March (1982) Vol. II, p. 707.

[12] Westerhof, E., *Implementation of TORAY at JET*, Rijnhuizen Report 89-183 (1989).

[13] Demeio, L., and Engelmann, F., *Plasma Phys. Contr. Fusion* **28** (1986) 1851.

[14] O'Brien, M.R., Cox, M., and Start, D.F.H., *Nucl. Fusion* **26** (1986) 1625.

[15] Kennel, C.F., and Engelmann, F., *Phys. Fluids* **9** (1966) 2377.

[16] Richtmyer, R.D., and Morton, K.W., *Difference Methods for Initial Value Problems*, 2nd ed., Interscience, New York, USA (1967).

[17] Marchuk, G.I., *Methods of Numerical Mathematics*, Springer Verlag, New York, USA (1975).

[18] Spitzer, L., *Physics of Fully Ionized Gases*, John Wiley and Sons, New York, USA (1962).

[19] Coppi, B., and Sigmar, D.J., *Phys. Fluids* **16**, (1973) 763.

[20] Karney, C.F.F., 31st Meeting of the Division of Plasma Physics of the APS, Anaheim, California, USA, Nov. 1989.

[21] Da Cruz, D.F., Peeters, A.G., Donné, A.J.H., Lopes Cardozo, N.J., and Westerhof, E., *Non-Maxwellian aspects of the electron distribution function in ECR heated tokamak plasma*, submitted for publication (1992).

[22] Giruzzi, G., *Phys. Fluids* **31** (1988) 3305.

[23] Cohen, R.H., *Phys. Fluids* **30** (1987) 2442 and **31** (1988) 421.

[24] Dongarra, J.J., Moler, C.B., Bunch, J.R., and Stewart, G.W., *LINPACK Users' Guide*, SIAM, Philadelphia, USA (1979).

[25] Sillen, R.M.J., Allaart, M.A.F., Goedheer, W.J., and Kattenberg, A., *NOTEC, a Code to Simulate ECE Spectra of Plasmas which Include Non-Thermal Populations*, Rijnhuizen Report 86-165 (1986).

[26] Freund, H.P., and Wu, C.S., *Phys. Fluids* **20** (1977) 963.